

# **SISTEMAS INFORMÁTICOS**

## **Curso 2012-2013**



# ***CAMINO SEGURO AL COLE***

**Aplicación móvil en colaboración con el Ayuntamiento de Madrid**

### **Autores:**

**Álvaro Bustillo Rebanal**

**Arturo Callejo Luis**

**Héctor A. Martos Gómez**

### **Directores:**

**Jesús Escribano Martínez**

**Victoria López López**

**Facultad de Informática  
Universidad Complutense de Madrid**



Álvaro Bustillo Rebanal, Arturo Callejo Luis y Héctor Alejandro Martos Gómez, alumnos matriculados en la asignatura de Sistemas Informáticos, autorizan a la Universidad Complutense de Madrid a difundir y utilizar con fines académicos, no comerciales y mencionando expresamente a sus autores, tanto la memoria como el código, los contenidos audiovisuales, la documentación y/o el prototipo desarrollado durante el curso académico 2012-2013 bajo la dirección de María Victoria López López, profesora del Departamento de Arquitectura de Computadores y Automática y Jesús Escribano Martínez, profesor del Departamento de Sistemas Informáticos y Computación.

Álvaro Bustillo Rebanal

Arturo Callejo Luis

Héctor A. Martos Gómez

Cita:

“The way you learn anything is that something fails, and you figure out how not to have it fail again”.

-Robert S. Arrighi, Pursuit of Power: NASA's Propulsion Systems Laboratory No.1 and 2

## Prólogo

*Camino Seguro al Cole* es un proyecto del Ayuntamiento de Madrid para fomentar la autonomía de los niños en su camino diario al colegio. Al poco tiempo de que Héctor, Álvaro y Arturo empezaran con este proyecto, pudieron darse cuenta de que no era tan sencillo como había parecido en un principio. No se trataba de una simple aplicación para Android. El producto resultante debía interaccionar con datos procedentes de cuatro Direcciones Generales relacionadas con el Área de Medio Ambiente y Movilidad y con dos Direcciones Generales relacionadas con Informática del Ayuntamiento de Madrid (IAM). Como consecuencia, los autores de este trabajo han tenido que participar activamente en reuniones de especificación con profesionales de diversos sectores y han realizado pruebas del producto desarrollado en el colegio Nuestra Señora de la Paloma, con resultados satisfactorios.

Como directores de este proyecto queremos destacar la magnífica actitud de este equipo en todo momento, su entusiasmo y su capacidad de trabajo, elementos que han sido clave para cumplir unos ambiciosos objetivos. Por todo ello creemos que este trabajo es merecedor de la calificación más alta.

Victoria López y Jesús Escribano

## Resumen

En esta memoria se describe en detalle la aplicación móvil *Camino Seguro al Cole*, una aplicación desarrollada para Android que surge como una ayuda al proyecto "Madrid a pie: camino seguro al cole" emprendido por el Ayuntamiento de Madrid en el año 2007. El objetivo de dicho proyecto es fomentar la autonomía de los niños de cierta edad en la ciudad, para lo que se han destinado numerosos recursos para mejorar la seguridad y la accesibilidad de los caminos que siguen la mayor parte de los escolares para ir al colegio. Este entramado de caminos es lo que se denomina "araña de movilidad", y fija el camino que deben seguir los niños para ir seguros al colegio. Este camino contará además con la vigilancia de los comerciantes de la zona que hayan querido inscribirse en el proyecto.

La aplicación *Camino Seguro al cole* se convierte en una herramienta de apoyo a la familia, con la que padres e hijos podrán trabajar en la autonomía de los más jóvenes en la ciudad. Esta herramienta ofrece la posibilidad de visualizar el entramado completo de todos los caminos seguros sobre un mapa, mejorando la orientación de los niños, que es una parte esencial en el proceso de adquisición de autonomía. La aplicación *Camino Seguro al Cole* ofrece también la posibilidad de guiar al niño desde su casa al colegio, pudiendo ver en todo momento su posición en el mapa, para que padres e hijos puedan realizar juntos el camino, dejando que el niño vaya siendo el guía y favoreciendo su movilidad en la ciudad.

Otra de las funcionalidades de la herramienta es la notificación de cualquier incidencia que haya en el camino seguro al cole directamente al Ayuntamiento de Madrid, haciendo a los niños partícipes del mantenimiento de la ciudad. También podrán hacer una valoración de los comercios adheridos al proyecto, valorando la atención y ayuda prestada por el comerciante. Por último, la aplicación supone una ayuda para la obtención de datos sobre los caminos que siguen los escolares desde su casa al colegio, permitiendo dibujar el trazado sobre un mapa y enviar esta información directamente al Ayuntamiento de Madrid, para la correcta actualización de la "araña de movilidad".

## **Abstract**

In this memory we will describe the mobile application “Safe way to the school”, an application developed for Android devices that supports the project “Walking Madrid: Safe way to the school”, started by Madrid’s city council in the year 2007. The objective of that project was to encourage children’s autonomy in the city. A big number of human resources have been destined to improve the security and accessibility of the ways that most of the children take to go to school. This group of streets is named “mobility spider”, and set the way that children must follow to go safe to the school. This way will be also guarded by the shop-keepers in the zone that support to the project.

The application “Safe way to the school” is a family tool, to help parents and children to work together in the autonomy. This tool affords children to see the full grid of safe ways to the school in a map, improving the orientation of children, a very important need in the process of improving autonomy. The application also allows children to be guided from their current position to the school, with simple instructions and seeing each moment their current position in the map. This is very useful for the children to learn with their parents how to go alone to the school and to handle themselves in the city.

Another of the functionalities of the application is the notification of any incidence in the safe way to the school directly to the city council, making child take care of the city. They could also make an evaluation of the shops attached to the project, estimating the care and help offered by the shop-keeper. Finally the application helps the city council in the process of taking the data of the ways that children use to take to go to school by directly marking it in a map. This drawn map will be sent to the city council in order to update the “mobility spider”.

# Índice

Capítulo 1.	Introducción	9
1.1.	Motivación	9
1.2.	Definición del proyecto	10
Capítulo 2.	Especificación de requisitos	17
2.1.	Descripción general	17
2.2.	Primera especificación de la herramienta	17
2.2.1.	Requisitos de usuario	17
2.2.2.	Requisitos de sistema	19
2.3.	Problemas de la primera especificación	23
2.4.	Segunda especificación de la herramienta	24
2.4.1.	Requisitos de usuario	24
2.4.2.	Requisitos de sistema	26
Capítulo 3.	Desarrollo de la aplicación	31
3.1.	Descripción general	31
3.2.	Evolución temporal	31
3.3.	Arquitectura del sistema	33
3.3.1.	Vista de Casos de uso	33
3.3.2.	Vista lógica	48
3.3.3.	Vista de procesos	52
3.3.4.	Vista de despliegue	53
3.3.5.	Vista de implementación	54
3.3.6.	Vista de datos	67
Capítulo 4.	Técnicas especiales utilizadas	69
4.1.	Fórmula de Haversine	69
Capítulo 5.	Conclusiones y trabajo futuro	73
5.1.	Conclusiones	73
5.2.	Futuras líneas de desarrollo	74
Bibliografía y Referencias		77
Anexos		79
Anexo 1.	Definiciones, acrónimos y abreviaturas	79
Anexo 2.	Distribución actual de las versiones de Android	80
Anexo 3.	Contenido de las páginas PHP referenciadas desde la clase Llamada	82
Anexo 4.	Manual de usuario	89
Anexo 5.	Manual de administración	100
Anexo 6.	Registro de reuniones de especificación	115

# Capítulo 1. Introducción

## 1.1. Motivación

*Camino seguro al cole* es un proyecto del Ayuntamiento de Madrid que se inició en el año 2007, con el objetivo de favorecer y facilitar la autonomía de los niños de cuarto, quinto y sexto de primaria, proporcionándoles un camino seguro al colegio con la vigilancia de la Policía Municipal y los comerciantes adheridos al proyecto.

Han estado implicadas en el desarrollo del proyecto cuatro áreas de gobierno (Medio Ambiente, Seguridad y Movilidad, Familia y Servicios Sociales, y Obras y Espacios Públicos), nueve direcciones generales y los servicios de educación de las juntas municipales de distrito. Todos ellos han participado en la orientación e implantación en los centros escolares, la difusión entre los agentes sociales del barrio, la limpieza de los accesos a los centros escolares, la seguridad vial y la renovación y adecuación de calles. De este proyecto surge otro proyecto, la creación de una aplicación móvil de apoyo a las familias para que los padres ayuden a sus hijos a fomentar su autonomía en la ciudad, acompañándoles en el camino al colegio y enseñándoles a ir solos.

La aplicación ofrece la posibilidad de visualizar sobre un mapa el camino seguro al colegio, calculado a partir de una araña de movilidad para cada colegio, que contiene el entramado de calles que forman el camino seguro.

También ofrece la posibilidad de guiar al usuario con sencillas instrucciones localizando en cada momento su posición actual en el mapa. La aplicación *Camino seguro al cole* también refuerza la colaboración de los niños en las ciudades, permitiendo notificar fácilmente cualquier incidencia que haya en el camino al colegio, pudiendo adjuntar una fotografía de la incidencia. Esta información será remitida al Ayuntamiento de Madrid. Otra de las funcionalidades que ofrece *Camino seguro al cole* es hacer una valoración de un comercio adherido a la promoción, valorando la ayuda prestada por el comerciante y la atención recibida.

Las arañas de movilidad pueden sufrir modificaciones con el tiempo, por eso, la aplicación ofrece la posibilidad de recopilar los datos sobre las modificaciones en los caminos para generar o actualizar el camino seguro al colegio. Permite al usuario elegir y ver sobre un mapa el camino que sigue desde su casa al colegio y enviar esta información para la generación de la araña de movilidad. Esta tarea se realizaba a mano, los niños dibujaban el camino desde su casa al colegio en un mapa impreso en papel, por lo que la aplicación *Camino seguro al cole* permite la automatización de este proceso.

*Camino seguro al cole* es una aplicación innovadora, no existe ninguna aplicación que reúna todas sus funcionalidades, aunque sí hay aplicaciones similares, que se comentarán en el segundo capítulo. En los siguientes capítulos se



desarrollan los aspectos más específicos de la aplicación. El segundo capítulo trata la especificación de requisitos, en el que se detalla las necesidades para el desarrollo de la aplicación. En el tercer capítulo se realiza una descripción detallada del diseño, explicando las técnicas especiales utilizadas en el cuarto capítulo. En el quinto capítulo se exponen las conclusiones finales seguido de la bibliografía y los anexos.

## 1.2. Definición del proyecto

Como se especificaba en la sección anterior, la aplicación *Camino seguro al cole* no tiene competidores directos. Vamos a comentar las aplicaciones más significativas que tengan alguna funcionalidad común con la aplicación *Camino seguro al cole*.

### Gasolineras España



#### Descripción:

Esta aplicación permite consultar las estaciones de servicio cerca de la ubicación del usuario, pudiendo ordenarlas por precio o distancia y mostrarlas

sobre un mapa. También permite ver el precio del combustible (gasolina, diesel, biodiesel) en cada una de las estaciones de servicio para buscar las más baratas.

*Gasolineras España*<sup>[1]</sup> es una aplicación que hace uso del sistema de localización GPS y la visualización de la información sobre un mapa de Google Maps, haciendo uso de la API v1. La información sobre las estaciones de servicio las obtiene de una base de datos propiedad del Ministerio de Industria. Por todas estas similitudes con la aplicación *Camino seguro al cole* es por lo que hemos elegido esta aplicación para comentarla.

El código de colores con el que cuenta la aplicación hace que el uso sea muy intuitivo, indicando en color verde las gasolineras más baratas y en rojo las más caras. Pulsando sobre los marcadores del mapa accedemos a más información sobre la estación de servicio seleccionada. Dentro de la vista de detalle hay una opción para guiarnos desde la posición en la que nos encontramos hasta la estación de servicio que hayamos seleccionado, mostrando la distancia a la que estamos. Esta opción abre la aplicación Google Maps Navigation, introduciendo como punto de origen la ubicación actual y como punto destino la dirección de la estación de servicio.

Otra de las opciones que ofrece la aplicación es listar las gasolineras por precio o por distancia desde la ubicación actual. También tiene una lista de favoritas, a la que podemos añadir cualquier estación pulsando sobre la estrella en la pantalla de detalle. Tenemos un menú de preferencias, en el que podemos fijar algunos parámetros como la capacidad del depósito de nuestro coche y el número máximo de estaciones a listar.

## Farmacias Ahora! Zaragoza



### Descripción:

La aplicación *Farmacias Ahora! Zaragoza*<sup>[2]</sup> te permite saber qué farmacias están abiertas en el momento de la búsqueda. Hace una búsqueda en una base de datos para comparar los horarios de apertura de las diferentes farmacias con la hora actual. Es una aplicación para el uso sobre una región (Zaragoza), algo que comparte con la aplicación *Camino seguro al cole*, que es una aplicación destinada al uso en Madrid.

La aplicación permite consultar toda la información de las farmacias que están abiertas en el momento de la búsqueda (teléfono, dirección, horario habitual), así como proporcionar instrucciones para llegar a la farmacia que seleccionemos como destino. Proporciona instrucciones para llegar en coche o a pie, que es algo que no ofrecía la anterior aplicación por falta de sentido y que es básico en la aplicación *Camino seguro al cole*. La aplicación *Farmacias Ahora! Zaragoza* no usa los mapas de Google Maps, aunque sí que usa las indicaciones que proporciona Google Maps para llegar desde el origen hasta el

destino. Los datos de las farmacias se obtienen de una base de datos propiedad del Colegio Oficial de Farmacéuticos de Zaragoza.

Como opciones adicionales permite llamar directamente al teléfono proporcionado en la información de la farmacia, listar las farmacias por distancia al usuario y mostrar un listado de teléfonos de interés.

## SAFA (Sagrada Familia)



### Descripción:

SAFA<sup>[3]</sup> es una aplicación muy sencilla que nos permite conocer información sobre los centros educativos de la Fundación SAFA. Podemos ver la oferta educativa, un mapa con los centros educativos y ser guiados hasta el colegio que seleccionemos vía GPS. Esta aplicación ha sido incluida en la definición del proyecto por las similitudes que tiene con la aplicación Camino seguro al cole, ya que las dos están orientadas a la localización de un colegio en un mapa.

Como opciones adicionales, permite acceder a la página web de la fundación, hacer una llamada a alguno de los centros educativos y mandar un email para cualquier duda o sugerencia.

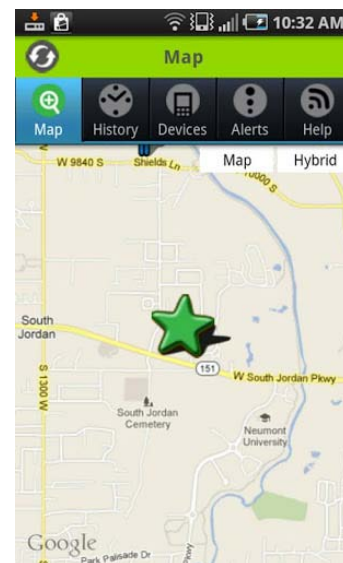
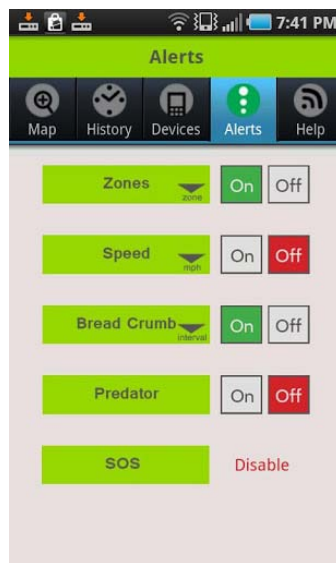
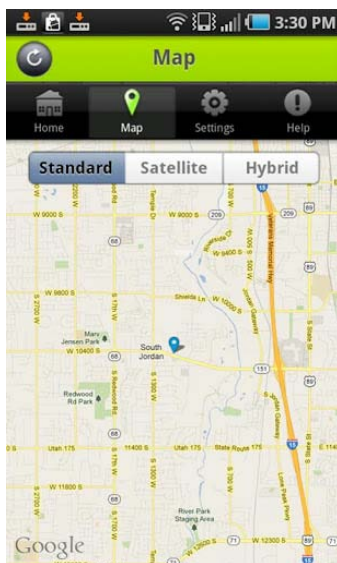
## Amber GPS Alert



### The Newest Innovation in Child Location GPS Systems

Start Protecting Your Child Today

Buy Now



### Descripción:

*Amber GPS Alert*<sup>[4]</sup> es una aplicación de rastreo de dispositivos móviles, está formada por dos aplicaciones diferentes, una para el niño a rastrear (*Amber GPS Alert Teen*) y otra para el adulto a cargo del niño (*Amber GPS Alert Parent*). La aplicación *Amber GPS Alert* es una aplicación orientada a la seguridad de los menores, en cualquier trayecto que pueda realizar, por lo que la hemos incluido en la definición del proyecto.

La aplicación hace uso de la API v1 de Google Maps para la representación de la localización del usuario sobre un mapa. Permite fijar una zona sobre el mapa y que se envíe un aviso a la aplicación del adulto responsable del niño si éste entra/sale de esa zona, además del envío de notificaciones periódicas de la localización exacta del niño. También cuenta con un botón de aviso de socorro, para situaciones de emergencia, con notificación a la aplicación del adulto.

La aplicación permite fijar un límite de velocidad, y enviar una notificación a la aplicación del responsable si se supera dicho límite; ésta es una manera de saber si el niño se monta en algún vehículo. La aplicación también consulta el registro nacional de agresores sexuales de Estados Unidos y envía



un aviso si el niño se acerca a menos de 500 metros de la residencia de alguno (sólo en Estados Unidos).

Como opciones adicionales permite la comunicación por voz entre las aplicaciones del niño y el adulto, además del envío de una notificación cuando el dispositivo del niño tenga menos de un 15% de batería.

## Waze



### Descripción:

Waze<sup>[5]</sup> es una aplicación social de navegación; es un sistema de navegación GPS que cuenta con una comunidad de 30 millones de usuarios que pueden comunicarse para intercambiar información del tráfico, permitiendo al usuario cambiar la ruta en tiempo real. El intercambio de información se realiza mediante marcadores en un mapa compartido para avisar de atascos, accidentes, radares y otros eventos. También

permite la comunicación de los usuarios mediante un chat, y notificaciones cuando tus amigos se conecten a Facebook.

Se puede compartir el precio del combustible en cualquier gasolinera, ayudando a otros usuarios de la comunidad a ahorrar. Cuenta con un sistema de reconocimiento vocal y recálculo automático de ruta si cambian las condiciones del tráfico.

## **Capítulo 2. Especificación de Requisitos**

### **2.1. Descripción general**

El propósito de esta sección es describir las necesidades y dificultades que se han tenido durante el desarrollo de la aplicación *Camino seguro al cole*. Se expondrán los requisitos de usuario y de sistema de las diferentes versiones de la aplicación, comentadas por secciones. Se comentarán los problemas que se han tenido en la especificación de la herramienta y las soluciones que se han propuesto.

### **2.2. Primera especificación de la herramienta**

#### **2.2.1. Requisitos de usuario**

La aplicación *Camino seguro al cole* se planteó como una aplicación para los niños de cuarto, quinto y sexto de primaria. La aplicación constaba de las siguientes funcionalidades:

1. Registrar e identificar un usuario en la aplicación, permitiendo su acceso al resto de funcionalidades.
2. Consultar la araña de movilidad para el colegio seleccionado previamente en el registro sobre un mapa.
3. Guiar al niño hasta el camino seguro al cole, con sencillas instrucciones y visualizando en todo momento su posición sobre un mapa.
4. Ver una lista con la información de todos los comercios amigos de los niños en la zona del colegio seleccionado en el registro.
5. Notificar una incidencia en el camino seguro al cole al Ayuntamiento de Madrid.
6. Envío de un mensaje al responsable del niño cuando éste haya llegado correctamente al colegio.
7. Aviso de emergencias al responsable del niño y a los comerciantes de la zona adscritos al proyecto.



### Requisitos:

El usuario debe completar correctamente el formulario de registro, rellenando al menos los campos obligatorios (número de teléfono, contraseña, colegio y número de teléfono del padre/madre/tutor).

El usuario debe acceder al sistema con los datos proporcionados en el registro para poder acceder al resto de funcionalidades de la aplicación.

El usuario debe activar los servicios de localización de su dispositivo si va a hacer uso de la funcionalidad de guiarle hasta el camino seguro.

Diagrama de flujo de la aplicación:

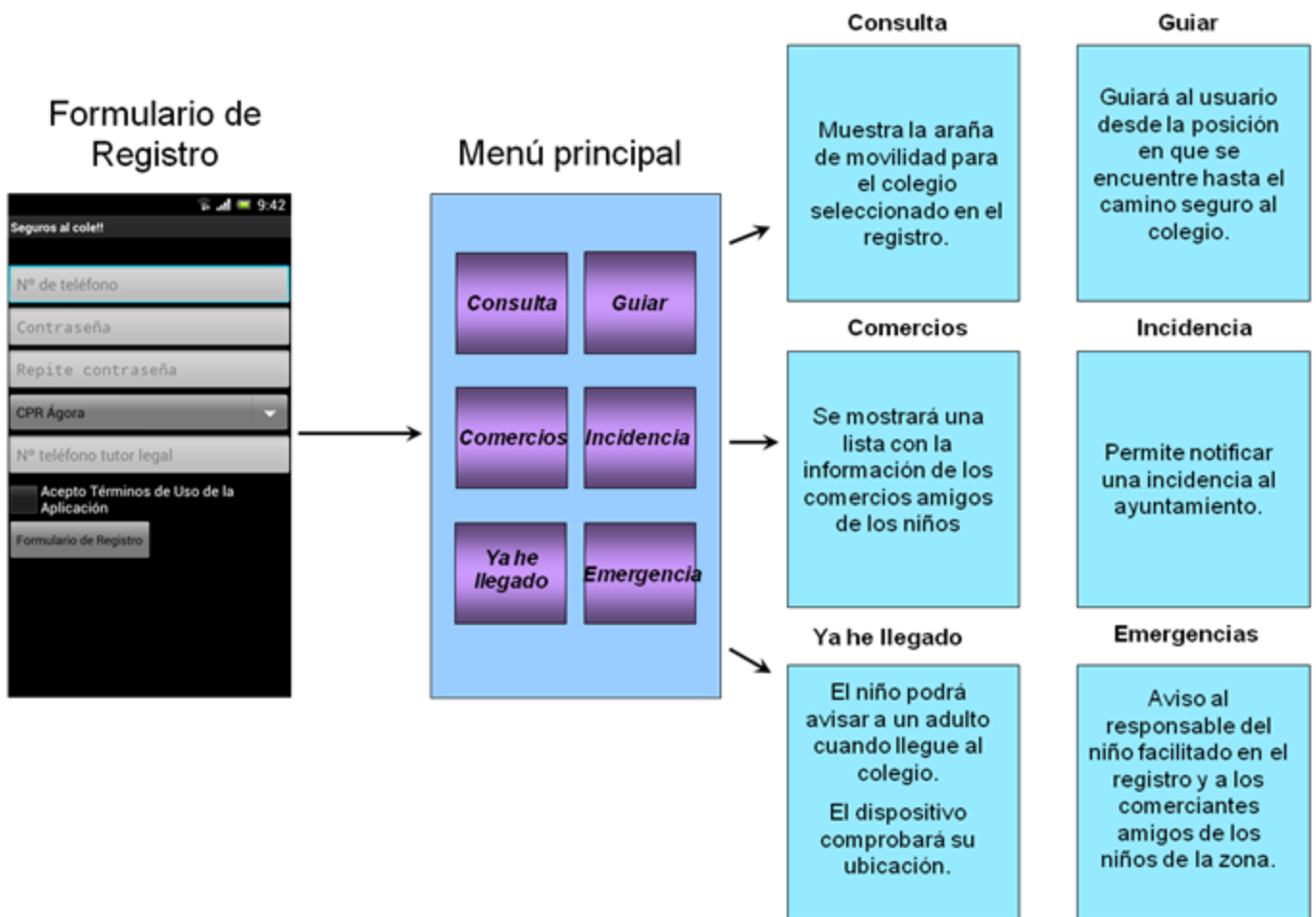


Figura 1: Diagrama de flujo de la primera especificación de la herramienta.

## **2.2.2. Requisitos de sistema**

### **Requisitos funcionales**

#### **Funcionalidad:**

Registro de un usuario en la aplicación. Una vez registrado, el usuario podrá acceder al resto de funcionalidades de la aplicación introduciendo en la pantalla de acceso los datos número de teléfono y contraseña con los que hizo el registro.

#### **Requisitos:**

El sistema debe disponer de una conexión activa de datos.

El sistema debe almacenar los datos del registro de usuario.

El sistema debe poder consultar los datos de registro de usuario para compararlos con los datos de acceso.

El sistema debe almacenar localmente los datos de acceso del usuario, de forma que no tenga que introducirlos cada vez que entre en la aplicación.

#### **Funcionalidad:**

Consultar la araña de movilidad, mostrando sobre un mapa el entramado de caminos seguros al colegio.

#### **Requisitos:**

El sistema debe almacenar la araña de movilidad.

El sistema debe ser capaz de recuperar la información de la araña de movilidad para mostrarla sobre un mapa.

El sistema debe disponer de una conexión activa de datos.

#### **Funcionalidad:**

Guiar al usuario desde la posición en la que se encuentre hasta el camino seguro al cole.

#### **Requisitos:**

El sistema debe ser capaz de obtener la localización GPS del dispositivo, para poder mostrar su posición actual sobre el mapa y para guiarle hasta el camino seguro.

El sistema debe disponer de una conexión activa de datos.

#### **Funcionalidad:**

Mostrar una lista con la información de los comercios amigos de los niños asignados al colegio con el que se registró el usuario.

#### **Requisitos:**

El sistema debe poder almacenar y consultar la información relativa a los comercios asignados a cada colegio.

El sistema debe disponer de una conexión activa de datos.

**Funcionalidad:**

Notificar una incidencia que haya en el camino seguro al cole al Ayuntamiento de Madrid

**Requisitos:**

El sistema debe ser capaz de acceder a la sección de envío de quejas y sugerencias de la página web del Ayuntamiento de Madrid.

El sistema debe disponer de una conexión activa de datos.

**Funcionalidad:**

Aviso de llegada al colegio.

**Requisitos:**

El sistema debe ser capaz de obtener la localización GPS del dispositivo para comprobar que se encuentra en un área cercana al colegio.

El sistema debe ser capaz de enviar una notificación textual al teléfono del padre/madre/tutor del usuario proporcionado durante el registro.

El sistema debe disponer de una conexión activa de datos.

**Funcionalidad:**

Aviso de emergencias al responsable del usuario y a los comercios amigos de los niños.

**Requisitos:**

El sistema debe ser capaz de obtener la localización GPS actual.

El sistema debe ser capaz de enviar una notificación textual acompañada de la posición actual del dispositivo al teléfono del padre/madre/tutor del usuario proporcionado durante el registro.

El sistema debe ser capaz de acceder a la información de los comercios para enviarles al teléfono la misma notificación que al responsable del usuario.

El sistema debe disponer de una conexión activa de datos.

## **Requisitos no funcionales**

### **Usabilidad**

La aplicación está destinada al uso de niños de cuarto, quinto y sexto de primaria, por lo que deberá ser fácil de usar. Los menús deben ser sencillos, con botones grandes, y se debe permitir volver atrás con el botón atrás del dispositivo.

Los movimientos posibles son la pulsación, el arrastre, y todos los gestos que permite la API de Google Maps sobre los mapas (ampliar, reducir y aumentar en una zona).

### **Fiabilidad**

Pruebas de fiabilidad realizadas sobre el terminal Sony Xperia Neo V(MT11i), LG Optimus Black(LG P-970) y Thc.

Se pondrá a prueba la aplicación en diferentes situaciones, y se comprobará que se resuelven exitosamente.

### **Rendimiento**

Mediremos el rendimiento de la aplicación sobre los mismos terminales que se usan en las pruebas de fiabilidad.

### **Restricciones de diseño**

#### **Uso del entorno de desarrollo Eclipse**

Para la implementación de la aplicación usaremos el entorno de desarrollo Eclipse, versión Helios, junto con el kit de desarrollo de aplicaciones Android.

#### **Uso del SDK de Android 2.3**

Desarrollaremos la aplicación sobre la versión 2.3 Gingerbread de Android (API 10), versión de Android más usada en Febrero del 2013<sup>1</sup>.

### **Componentes usados**

#### **Uso de la API de Google Maps v1**

Para mostrar la ubicación del usuario en el mapa y para guiarle hasta el camino seguro al cole usamos la API de Google Maps v1, de uso gratuito hasta 2.500 solicitudes diarias.

---

<sup>1</sup> Según las estadísticas publicadas por Google Inc, ver Anexo 2. Distribución actual de las versiones de Android.

### **Uso de la API de Whatsapp**

Para el envío de notificaciones al responsable del usuario y a los comerciantes de la zona adscritos al proyecto *Camino seguro al cole*. Habrá que estudiar si es posible usar la API<sup>2</sup> gratuitamente.

## **Interfaces**

### **Interfaces de usuario**

En líneas generales, en todas las interfaces gráficas predominan los colores vivos, que dan una visión más atractiva a los usuarios a los que está destinado, escolares comprendidos entre 10 y 12 años. Se buscará que las interfaces sean intuitivas y simples, facilitando así su manejo.

Se mostrarán mensajes informativos al usuario durante el uso de la aplicación. El usuario podrá pulsar en cualquier momento el botón de ayuda situado en la esquina superior derecha de la pantalla.

### **Interfaces software**

Interfaz de la API de Google Maps v1 para mostrar las arañas de movilidad sobre el mapa y para guiar al usuario hasta el camino seguro al cole. Las acciones que permite la interfaz son ampliar y reducir el zoom, movernos arrastrando el dedo por el mapa y ampliar a una determinada zona tocando dos veces seguidas el mismo punto en la pantalla.

Interfaz del navegador elegido para abrir la página de notificaciones del Ayuntamiento.

Interfaz de Whatsapp para la recepción de notificaciones de emergencias.

### **Interfaces hardware**

No se usa ninguna interfaz hardware adicional.

---

<sup>2</sup> La utilización de esta API ya no es necesaria, como se describe más adelante.

## 2.3. Problemas de la primera especificación

### **Problema:**

Los usuarios a los que está destinada la aplicación son niños de cuarto, quinto y sexto de primaria, así que tendrán entre diez y doce años. En este rango de edad no es frecuente que los niños tengan un móvil que cumpla con los requisitos hardware especificados en la sección correspondiente. Además, los colegios no permiten el uso de teléfonos móviles a los alumnos en este rango de edad.

### **Solución:**

Convertimos la aplicación en una herramienta familiar, con la que padres e hijos puedan trabajar conjuntamente en la autonomía de los menores. Se trata de que los padres enseñen a sus hijos a ir al colegio utilizando la ayuda que proporciona la aplicación *Camino seguro al cole* instalada en sus dispositivos. La especificación inicial cambia, y se elimina la funcionalidad de avisar de una emergencia, ya que con la nueva visión del proyecto los padres acompañarán a los menores en el uso de la aplicación. Se elimina también la funcionalidad de avisar cuando el niño llegue al colegio, ya que no será él el que lleve el dispositivo. Se añade la funcionalidad de valorar la ayuda recibida por parte de un comercio amigo de los niños, extendiendo la funcionalidad de mostrar la información de los comercios.

### **Problema:**

La API de la aplicación Whatsapp es privada, por lo que no podemos hacer uso de la aplicación para el envío de notificaciones.

### **Solución:**

El problema queda resuelto con el cambio en la especificación, ya que se elimina la funcionalidad de aviso de emergencias, que era la funcionalidad que requería del envío de notificaciones.

### **Problema:**

Las arañas de movilidad de los colegios no permanecen inalteradas en el tiempo, sino que pueden sufrir modificaciones cada cierto tiempo. La tarea de recogida de datos y la introducción de esos datos en el sistema se hace manualmente, y es muy costosa en tiempo y recursos.

### **Solución:**

Añadimos una nueva funcionalidad a la herramienta, la posibilidad de que el usuario marque el camino que sigue desde su casa al colegio sobre un mapa en el dispositivo, y que el dispositivo automáticamente recoja esa información y la envíe al Ayuntamiento para su posterior tratamiento. Esto facilitará la tarea de actualización de la araña de movilidad.

### **Problema:**

La API de Google Maps se actualiza a la versión v2 el 3 de diciembre del 2012, dejando de proporcionar licencias de uso para la versión v1.

**Solución:**

Cambiamos el código implementado hasta el momento usando las instrucciones recogidas en la nueva API de Google Maps v2.

## **2.4. Segunda especificación de la herramienta**

### **2.4.1. Requisitos de usuario**

En la segunda especificación de la aplicación *Camino seguro al cole* hay un cambio en los actores. En la primera especificación el actor era un niño con una edad comprendida entre diez y doce años. En la segunda especificación hay dos actores que actúan conjuntamente; un niño entre diez y doce años y un adulto responsable del niño. En la segunda especificación de la herramienta se recogen las siguientes funcionalidades:

1. Registrar e identificar un usuario en la aplicación, permitiendo su acceso al resto de funcionalidades.
2. Consultar la araña de movilidad para el colegio seleccionado previamente en el registro sobre un mapa.
3. Guiar a los actores hasta el camino seguro al cole, visualizando en todo momento su posición sobre un mapa.
4. Hacer una valoración de la ayuda prestada por un comercio amigo de los niños en la zona del colegio seleccionado en el registro.
5. Notificar una incidencia en el camino seguro al cole al Ayuntamiento de Madrid.
6. Envío de rutas para la actualización de la araña de movilidad. El usuario trazará la ruta que sigue desde su casa hasta el colegio sobre un mapa. Esta información se envía directamente al Ayuntamiento de Madrid.

**Requisitos:**

Los usuarios deben completar correctamente el formulario de registro, rellenando al menos los campos obligatorios (número de teléfono, contraseña, colegio y número de teléfono del padre/madre/tutor).

Los usuarios deben acceder al sistema con los datos proporcionados en el registro para poder acceder al resto de funcionalidades de la aplicación.

Los usuarios deben activar los servicios de localización de su dispositivo si va a hacer uso de la funcionalidad de guiarle hasta el camino seguro.

Diagrama de flujo de la aplicación:

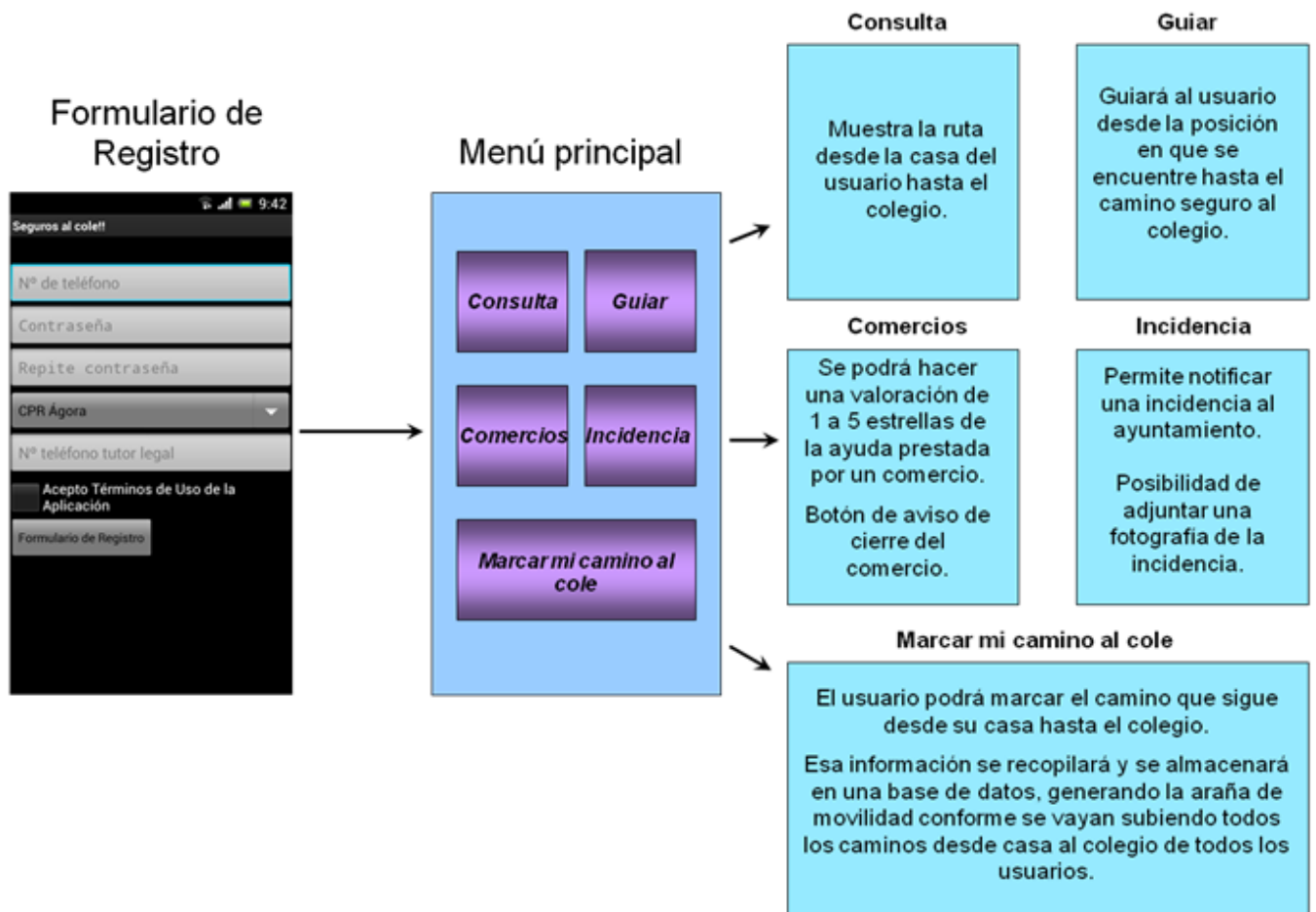


Figura 2: Diagrama de flujo de la segunda especificación de la herramienta.



## **2.4.2. Requisitos de sistema**

### **Requisitos funcionales**

#### **Funcionalidad:**

Registro de un usuario en la aplicación. Una vez registrado, el usuario podrá acceder al resto de funcionalidades de la aplicación introduciendo en la pantalla de acceso los datos número de teléfono y contraseña con los que hizo el registro.

#### **Requisitos:**

El sistema debe almacenar los datos del registro de usuario.

El sistema debe poder consultar los datos de registro de usuario para compararlos con los datos de acceso.

El sistema debe almacenar localmente los datos de acceso del usuario, de forma que no tenga que introducirlos cada vez que entre en la aplicación.

El sistema debe disponer de una conexión activa de datos.

#### **Funcionalidad:**

Consultar la araña de movilidad, mostrando sobre un mapa el entramado de caminos seguros al colegio.

#### **Requisitos:**

El sistema debe almacenar la araña de movilidad.

El sistema debe ser capaz de recuperar la información de la araña de movilidad para mostrarla sobre un mapa.

El sistema debe disponer de una conexión activa de datos.

#### **Funcionalidad:**

Guiar al usuario desde la posición en la que se encuentre hasta el camino seguro al cole.

#### **Requisitos:**

El sistema debe ser capaz de obtener la localización GPS del dispositivo, para poder mostrar su posición actual sobre el mapa y para guiarle hasta el camino seguro.

El sistema debe disponer de una conexión activa de datos.

#### **Funcionalidad:**

Hacer una valoración de la ayuda prestada por un comercio amigo de los niños en la zona del colegio seleccionado en el registro. Aviso del cierre de algún comercio.

**Requisitos:**

El sistema debe poder almacenar y consultar la información relativa a los comercios asignados a cada colegio, añadiendo un campo a la información que lleve la valoración media para cada comercio.

El sistema debe restringir la valoración múltiple de un usuario a un mismo comercio.

El sistema debe disponer de una conexión activa de datos.

**Funcionalidad:**

Notificar una incidencia que haya en el *Camino seguro al cole* al Ayuntamiento de Madrid

**Requisitos:**

El sistema debe ser capaz de acceder a la sección de envío de gestiones y trámites de la página web del Ayuntamiento de Madrid.

El sistema debe disponer de una conexión activa de datos.

**Funcionalidad:**

Envío de rutas para la actualización de la araña de movilidad.

**Requisitos:**

El sistema debe ser capaz de almacenar y enviar la información de la ruta trazada por el usuario sobre un mapa.

El sistema debe restringir el envío de más de una ruta por parte del mismo usuario.

El sistema debe disponer de una conexión activa de datos.

## **Requisitos no funcionales**

### **Usabilidad**

La aplicación está destinada al uso conjunto de padres e hijos, por lo que será una aplicación sencilla, en la que el niño pueda manejarse con autonomía, pudiendo pedir ayuda al padre si es necesario. Todas las funcionalidades de la aplicación están reunidas en un menú principal, sencillo y con botones grandes. Se permite volver al menú desde cualquier funcionalidad pulsando el botón atrás en el dispositivo. Los menús de opciones de las funcionalidades se despliegan pulsando el botón de opciones del dispositivo.

Los movimientos posibles son la pulsación, el arrastre, y todos los gestos que permite la API de Google Maps sobre los mapas (ampliar, reducir y aumentar en una zona).

### **Fiabilidad**

La aplicación no debe colapsarse durante su uso, ni forzar el cierre de la misma en ningún momento.

Se pondrá a prueba la aplicación en diferentes situaciones, y se comprobará que se resuelven exitosamente.

Pruebas de fiabilidad realizadas sobre el terminal Sony Xperia Neo V(MT11i), LG Optimus Black(LG P-970) y Thc.

### **Rendimiento**

La API de Google Maps tiene una licencia de carácter gratuito para menos de 25.000 accesos diarios, luego ese será el límite de usuarios fijado. Contando con un máximo de 25 alumnos por clase<sup>[6]</sup> y suponiendo un total de 3 grupos por curso, en los cursos 4º, 5º y 6º de primaria, aplicado a los 19 colegios que actualmente forman parte del proyecto; sale un total de 4275 usuarios, con lo que parece suficiente el límite de 25.000 accesos diarios, permitiendo la ampliación del proyecto a más colegios. Si se quisiera ampliar dicho límite tendrá que adquirirse una licencia Google Maps for Business, que permite 100.000 accesos diarios.

Debido al tráfico de datos es aconsejable disponer de una buena velocidad de descarga, aunque es posible el uso de la aplicación con un rendimiento menor. El mínimo de velocidad de descarga debe ser al menos de 64Kbps, siendo una velocidad recomendada 7,2Mbps. El rendimiento se ha medido sobre los mismos terminales especificados en la sección anterior con una conexión de datos activa con una velocidad de descarga de 7,2 Mbps.

## **Restricciones de diseño**

### **Uso del entorno de desarrollo Eclipse**

Para la implementación de la aplicación usaremos el entorno de desarrollo Eclipse, versión Juno, junto con el kit de desarrollo de aplicaciones Android SDK.

### **Uso del SDK de Android 4.0**

Desarrollaremos la aplicación sobre la versión 4.0 Ice Cream Sandwich de Android (API 15). Esta restricción se añade tras la actualización de la API de Google Maps a la versión v2. Esta versión de Android es la segunda más utilizada<sup>3</sup> y su uso va en aumento.

## **Componentes usados**

### **Uso de la API de Google Maps v2**

Para mostrar la ubicación del usuario en el mapa y para guiarle hasta el *Camino seguro al cole* usamos la API de Google Maps v2, de uso gratuito hasta 25.000 solicitudes diarias.

## **Interfaces**

### **Interfaces de usuario**

En líneas generales, en todas las interfaces gráficas predominan los colores vivos, que dan una visión más atractiva a los usuarios a los que está destinado, escolares comprendidos entre 10 y 12 años. Se buscará que las interfaces sean intuitivas y simples, facilitando así su manejo. El acceso a todas las funcionalidades se realiza a través del menú principal de la aplicación.

Se mostrarán mensajes informativos al usuario durante el uso de la aplicación.

---

<sup>3</sup> En el Anexo 2 se encuentran las estadísticas de uso de las distintas distribuciones de Android.

## **Interfaces software**

Interfaz de la API de Google Maps v2 para mostrar las arañas de movilidad sobre el mapa y para guiar al usuario hasta el *Camino seguro al cole*. Las acciones que permite la interfaz son ampliar y reducir el zoom, movernos arrastrando el dedo por el mapa y ampliar a una determinada zona tocando dos veces seguidas el mismo punto en la pantalla.

Interfaz del navegador elegido para abrir la página de notificaciones del Ayuntamiento.

## **Interfaces hardware**

No se usa ninguna interfaz hardware adicional.

## Capítulo 3. Desarrollo de la aplicación

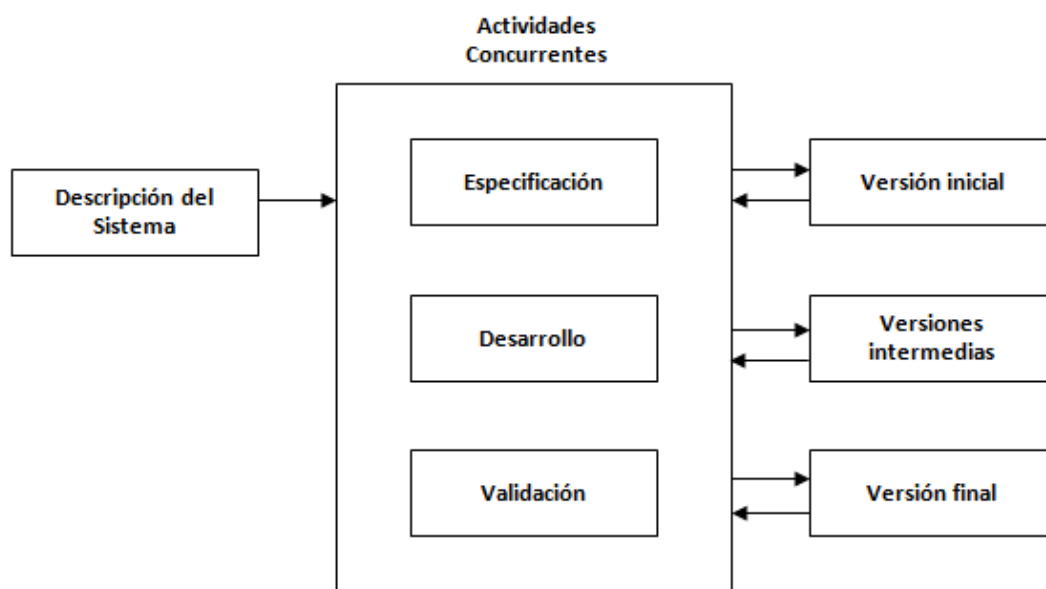
### 3.1. Descripción general

En este capítulo se realizará una descripción del diseño y la programación de la aplicación *Camino seguro al cole*. Se mostrará la evolución temporal del proyecto, con las acciones llevadas a cabo en cada periodo de tiempo representadas sobre un diagrama de Gantt.

Después se desarrollará la arquitectura del sistema, con las diferentes vistas, incluyendo diagramas UML que describan el comportamiento completo del sistema. Todos los diagramas están realizados con el programa Rational Software Architect, con lo que la notación podría diferir ligeramente del estándar UML 2.0.

### 3.2. Evolución temporal

El modelo de desarrollo software que hemos seguido ha sido el de desarrollo evolutivo, en el que se realizan concurrentemente las tareas de especificación, desarrollo y validación mientras se desarrollan sucesivas versiones de la aplicación. Este modelo de desarrollo busca reemplazar el viejo sistema con uno nuevo que tendría la propiedad de satisfacer los nuevos requerimientos lo más rápido posible. El desarrollo evolutivo asume que los requerimientos están sujetos a continuos cambios y ofrece en la implementación un reflejo, también continuo, de esos cambios.



En el siguiente diagrama de Gantt está representada la evolución temporal del proyecto, con todas las fases del modelo de desarrollo evolutivo repartidas por trimestres empezando desde el mes de octubre de 2012 y acabando en el mes de junio de 2013.

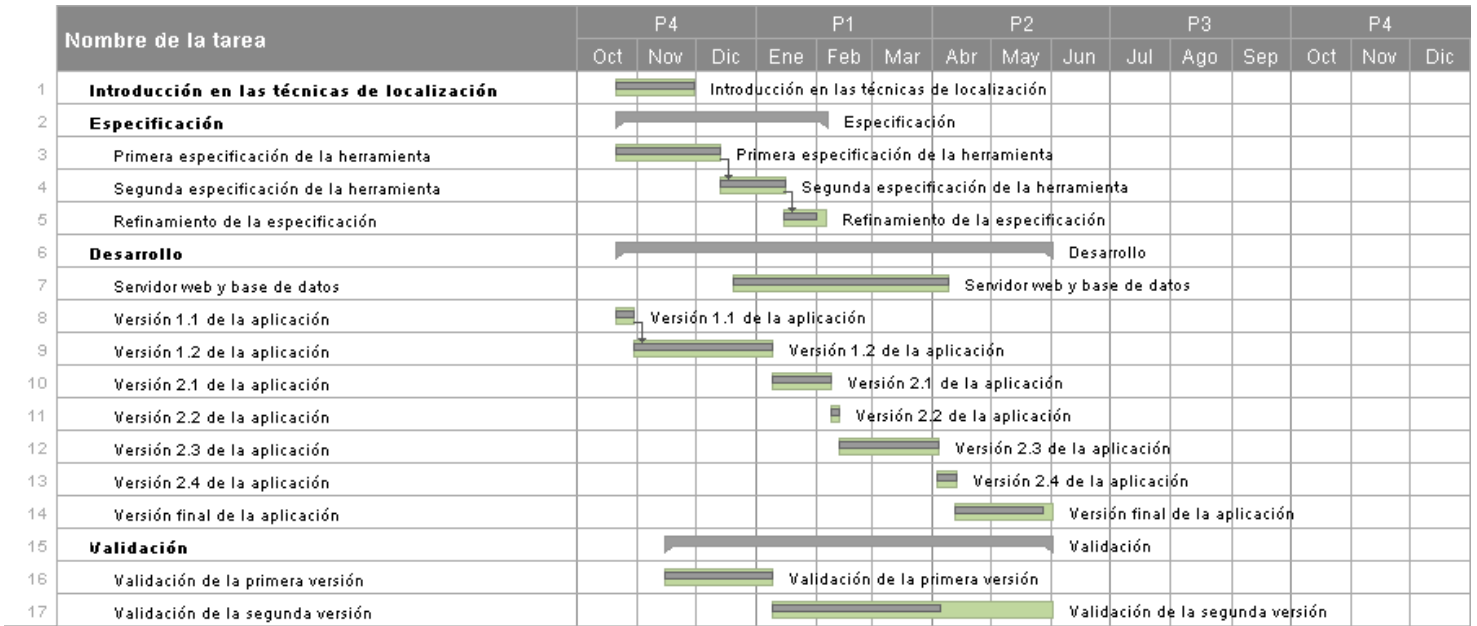


Figura 3: Diagrama de Gantt

### 3.3. Arquitectura del sistema

#### 3.3.1. Vista de casos de uso

En esta vista vemos el modelo de casos de uso de la aplicación. Tenemos dos actores principales, que son el usuario (un niño de cuarto, quinto o sexto de primaria) y un adulto responsable del niño, que tendrá la aplicación instalada en su móvil. Ambos actores actúan conjuntamente como si fueran uno.

El otro actor que aparece en el diagrama de casos de uso es el administrador, que se encargará de la gestión de los datos persistentes de la aplicación. Cada globo representa un fragmento de funcionalidad de la aplicación y las flechas indican la relación que hay entre ellas.

#### Diagrama de casos de uso

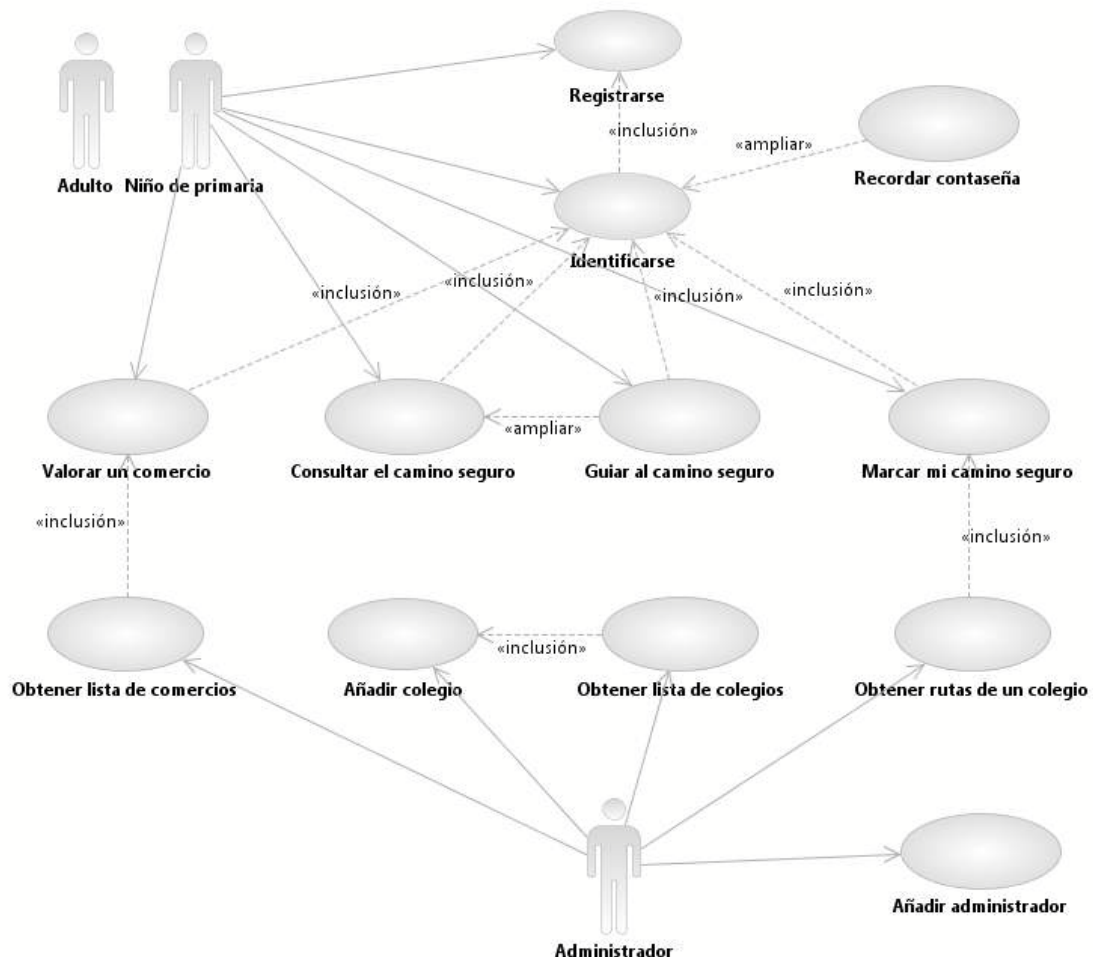


Figura 4: Diagrama de casos de uso.



## **Caso de uso: Registrarse en la aplicación**

### **Descripción breve:**

Este caso de uso describe la funcionalidad de la aplicación que permite al usuario registrarse en la aplicación para poder hacer uso de ella.

### **Flujo de eventos:**

#### **Camino principal:**

Este caso de uso comienza seleccionando la opción Registro en la pantalla inicial de la aplicación. El usuario deberá rellenar el formulario con información veraz. Se completarán todos los campos, a saber:

1. Número de teléfono móvil del usuario, desde el que el usuario usará la aplicación.
2. Contraseña de uso personal, puede contener letras y dígitos y debe tener una longitud mínima de cinco caracteres, haciendo distinción entre letras minúsculas y mayúsculas.
3. Repetición de la contraseña introducida anteriormente. Los dos campos deben coincidir.
4. Seleccionar el colegio del usuario de la lista de colegios inscritos en el proyecto *Camino seguro al cole*.
5. Código de seguridad del colegio.
6. Email, para la recuperación de la contraseña en caso de olvido.

Para completar el registro deberá leer y aceptar los términos de uso de la aplicación marcando la casilla *Acepto los términos de uso de la aplicación*.

Una vez completados todos los cambios y aceptados los términos de uso el usuario tendrá acceso al menú principal, desde el que podrá acceder a todas las funcionalidades de la aplicación.

#### **Caminos secundarios:**

Si alguno de los campos rellenos no es correcto se mostrará un mensaje informativo, y el usuario deberá subsanar los errores para finalizar el registro en la aplicación.

#### **Requisitos especiales:**

No debe haber ningún usuario registrado en la base de datos con el mismo número de teléfono.

#### **Precondiciones:**

Haber instalado correctamente la aplicación *Camino seguro al cole*.

#### **Postcondiciones:**

Los datos del usuario quedarán introducidos en la base de datos según especifiquen los términos de uso, y el usuario podrá identificarse usando estos datos para usar la aplicación

## Caso de uso: Identificarse en la aplicación

### Descripción breve:

Este caso de uso describe la funcionalidad de la aplicación que permite al usuario identificarse para acceder al resto de funcionalidades descritas en esta sección.

### Flujo de eventos:

#### Camino principal:

Para identificarse en la aplicación es necesario introducir los datos número de teléfono móvil del usuario y contraseña en la pantalla inicial. Estos datos deben coincidir con los datos que el usuario proporcionó en el registro de la aplicación.

#### Caminos secundarios:

El usuario tiene la opción de marcar la casilla *Recordarme* para que el dispositivo recuerde estos datos localmente y no tener que introducir los datos cada vez que entra en la aplicación.

El usuario tiene la opción de marcar la casilla *Entrar automáticamente* para que la aplicación haga la identificación de manera transparente al usuario.

Si el usuario ha olvidado su contraseña, puede pulsar sobre el botón *Olvidaste tu contraseña* para que se le envíe un email a la dirección que proporcionó en el registro con una nueva contraseña. Una vez enviada la nueva contraseña no podremos usar la antigua contraseña para identificarnos en la aplicación.

#### Requisitos especiales:

Para marcar la casilla *Entrar automáticamente* se debe haber marcado previamente la casilla *Recordarme*.

#### Precondiciones:

El usuario debe estar registrado en la aplicación, proporcionando correctamente estos datos en la identificación.

#### Postcondiciones:

El usuario queda identificado en la aplicación, pudiendo acceder a cualquiera de sus funcionalidades.

## Caso de uso: Consultar el camino seguro

### Descripción breve:

Este caso de uso describe la funcionalidad de la aplicación que permite al usuario consultar el *Camino seguro al cole*. El usuario podrá visualizar sobre un mapa el entramado de calles que forman la araña de movilidad.

### Flujo de eventos:

#### Camino principal:

Este caso de uso comienza seleccionando la opción *Mi ruta segura* en el menú principal de la aplicación. El usuario podrá visualizar sobre un mapa el entramado de calles que forman el *Camino seguro al cole*.

Este entramado de calles se mostrará añadiendo una línea verde superpuesta sobre las calles que forman la araña de movilidad para el colegio seleccionado.

Se mostrarán también el colegio y todos los establecimientos amigos de los niños asignados a ese colegio. El usuario podrá ampliar o reducir el zoom en el mapa y visualizar la información del colegio o de los establecimientos pulsando sobre los marcadores en el mapa.

#### Caminos secundarios:

Pulsando el menú de opciones del dispositivo se despliega un menú con 3 opciones:

- *Mostrar cómo llegar a la ruta segura*: descrito en el caso de uso *Guiar al colegio*.
- *Navegar libremente por el mapa*: permite moverse libremente por el mapa.
- *Fija el mapa sobre tu posición actual*: centra la vista del mapa sobre la posición actual del usuario.

### Requisitos especiales:

No hay requisitos especiales en este caso de uso.

### Precondiciones:

El usuario debe estar registrado e identificado en la aplicación.

El usuario debe haber seleccionado la opción *Mi ruta segura* en el menú principal de la aplicación.

### Postcondiciones:

Queda dibujada la araña de movilidad para el colegio que el usuario seleccionó en el registro.

## **Caso de uso: Guiar al colegio**

### **Descripción breve:**

Este caso de uso describe la funcionalidad de la aplicación que permite que el usuario sea guiado hasta el colegio seleccionado en el registro. Se trazará sobre el mapa el camino a pie más corto desde su posición actual hasta el colegio siguiendo la ruta segura fijada por la araña de movilidad.

### **Flujo de eventos:**

#### **Camino principal:**

Este caso de uso comienza seleccionando la opción *Mostrar cómo llegar a la ruta segura* del menú de opciones de la funcionalidad *Mi ruta segura*. A partir de este momento la aplicación recopilará datos de posicionamiento del dispositivo y calculará la ruta a pie más corta hasta el punto de entrada más cercano de la araña de movilidad.

#### **Caminos secundarios:**

Se mostrarán también el colegio y todos los establecimientos amigos de los niños asignados a ese colegio. El usuario podrá ampliar o reducir el zoom en el mapa y visualizar la información del colegio o de los establecimientos pulsando sobre los marcadores en el mapa.

### **Requisitos especiales:**

Para la obtención de la localización del usuario es necesario que éste haya activado previamente los servicios de localización GPS en el menú de ajustes de su dispositivo. Para ello aparecerá un mensaje desde el que podremos acceder a los ajustes de localización en el menú principal.

### **Precondiciones:**

El usuario debe estar registrado e identificado en la aplicación.

El usuario debe haber seleccionado la opción *Mi ruta* en el menú principal de la aplicación y haber seleccionado la opción *Mostrar cómo llegar a la ruta segura* del menú de opciones.

### **Postcondiciones:**

El usuario queda localizado en el mapa de Google, indicando su posición actual en el mapa con un marcador.

Se muestra con una línea azul superpuesta sobre las calles la ruta a pie hasta la araña de movilidad, y con una línea verde el entramado de calles de la araña de movilidad.

## **Caso de uso: Valorar un comercio**

### **Descripción breve:**

Este caso de uso describe la funcionalidad de la aplicación que permite al usuario dar una valoración de un comercio amigo de los niños. La valoración se hará en una escala de una a cinco estrellas, y el usuario ver las valoraciones medias de los comercios asignados al colegio seleccionado en el registro.

### **Flujo de eventos:**

#### **Camino principal:**

Este caso de uso comienza seleccionando la opción *Valorar comercios* en el menú principal de la aplicación. A continuación se listarán todos los comercios amigos de los niños que estén dentro de la araña de movilidad del colegio en el que esté inscrito el usuario.

La valoración del comercio se hará en una escala de una a cinco estrellas teniendo en cuenta la atención recibida por el comerciante.

#### **Caminos secundarios:**

El usuario puede modificar la valoración de un comercio en caso de que se haya equivocado o el comercio haya cambiado de dueño y quiera modificar su valoración

### **Requisitos especiales:**

Sólo se permitirá una valoración de cada comercio por usuario.

### **Precondiciones:**

El usuario debe estar registrado e identificado en la aplicación.

El usuario debe haber seleccionado la opción *Valorar comercios* en el menú principal de la aplicación.

El usuario no debe haber valorado el mismo comercio anteriormente.

### **Postcondiciones:**

La valoración quedará registrada en la base de datos, identificando al usuario que la ha realizado, para evitar que un usuario pueda hacer más de una valoración de un comercio.

## **Caso de uso: Notificar una incidencia**

### **Descripción breve:**

Este caso de uso describe la funcionalidad de notificar una incidencia que haya en el *Camino seguro al cole* directamente al Ayuntamiento de Madrid.

### **Flujo de eventos:**

#### **Camino principal:**

Este caso de uso comienza al pulsar el botón identificado como *Reportar incidencia* en el menú principal de la aplicación.

Se direcciona al usuario a la página de gestiones y trámites del Ayuntamiento de Madrid, para que notifique la incidencia.

#### **Caminos secundarios:**

Para abrir la página de gestiones y trámites del Ayuntamiento de Madrid, el dispositivo dará a elegir al usuario la aplicación con la que quiere abrir dicha página, manteniendo en segundo plano la ejecución de la aplicación *Camino seguro al cole*. Se abrirá entonces la aplicación seleccionada por el usuario para realizar la notificación de la incidencia.

Cuando se complete el proceso y se cierre la aplicación seleccionada para acceder a la página web se devolverá el control a la aplicación *Camino seguro al cole*.

### **Requisitos especiales:**

No hay requisitos especiales en este caso de uso.

### **Precondiciones:**

El usuario debe estar registrado e identificado en la aplicación.

El usuario debe haber seleccionado la opción *Notificar una incidencia* en el menú principal de la aplicación.

### **Postcondiciones:**

Queda notificada la incidencia directamente al Ayuntamiento de Madrid.

## **Caso de uso: Marcar mi camino al cole**

### **Descripción breve:**

Este caso de uso describe la funcionalidad de la aplicación que permite al usuario dibujar el camino que sigue normalmente desde su casa hasta el colegio sobre un mapa y subir esta información a una base de datos. Esto servirá para renovar las arañas de movilidad cuando sea necesario.

### **Flujo de eventos:**

#### **Camino principal:**

Este caso de uso comienza seleccionando la opción *Crear mi ruta al cole*. El usuario tendrá que ir seleccionando las calles que sigue desde su casa hasta el colegio eligiendo la calle que toma en cada intersección y pulsando sobre ella en el mapa. Esto trazará una línea de color azul entre cada uno de los puntos del mapa sobre los que pulse el usuario; siendo el primer punto la ubicación actual del usuario.

Una vez completado el camino hasta el colegio el padre/madre/tutor del usuario deberá revisar el camino seleccionado por el usuario y confirmar o descartar el trayecto.

Una vez confirmado el trayecto habrá que pulsar el botón de opciones del dispositivo y pulsar sobre la opción *Enviar*. El trayecto se incluirá en una base de datos para la actualización de la araña de movilidad<sup>4</sup>.

#### **Caminos secundarios:**

Si nos equivocamos en el dibujo de la ruta podemos pulsar el botón de opciones del dispositivo para desplegar un menú desde el que podemos deshacer el último tramo dibujado o empezar de nuevo la ruta.

Una vez completado el camino hasta el colegio el padre/madre/tutor del usuario deberá volver a revisar el camino seleccionado por el usuario y enviar la ruta.

### **Requisitos especiales:**

El primer punto de la ruta corresponde con la ubicación actual del usuario, luego para dibujar correctamente el camino desde su casa al colegio el usuario debe estar en su casa o en el colegio.

---

<sup>4</sup> La generación de la araña de movilidad está descrita en el Anexo 5. Manual de administración.

**Precondiciones:**

El usuario debe estar registrado e identificado en la aplicación.

El usuario debe haber activado la localización GPS en el menú de ajustes del dispositivo (puede acceder desde el menú emergente en el menú principal).

El usuario debe haber seleccionado la opción *Marcar mi camino al cole* en el menú principal de la aplicación.

**Postcondiciones:**

Queda introducido en la base de datos el camino del usuario desde su casa al colegio, comparándose con el camino de los demás usuarios y generándose la araña de movilidad para el colegio del usuario.



## Caso de uso: Obtener lista de comercios

### Descripción breve:

Este caso de uso describe la funcionalidad del panel de administración que permite a un administrador obtener un fichero con todos los comercios amigos de los niños asignados a cada colegio.

### Flujo de eventos:

#### Camino principal:

Este caso de uso comienza accediendo desde la página de administración<sup>5</sup> <http://www.segurosalcole.comoj.com/administracion/login.php>.

En esta página debemos acceder con el usuario y la contraseña de administrador.

Una vez identificados accedemos al panel de administración desde el que tendremos que seleccionar la opción *Obtener un registro con todos los establecimientos adheridos*.

Tendremos que elegir dónde guardar el fichero, que será un archivo Excel que contendrá los siguientes campos:

- Nombre del establecimiento
- Dirección
- Coordenadas en LatLng
- Nombre del colegio al que está asignado
- Valoración media

#### Caminos secundarios:

No hay caminos secundarios en este caso de uso.

### Requisitos especiales:

Para la correcta visualización del fichero deberá disponer de una versión de Microsoft Office Excel 2003 o superior.

### Precondiciones:

El actor (administrador) debe estar registrado previamente como administrador, y recordar su usuario y contraseña.

El administrador debe haber seleccionado la opción *Obtener un registro con todos los establecimientos adheridos* del panel de administración.

### Postcondiciones:

Queda en posesión del administrador un fichero con la información almacenada en la base de datos de todos los comercios amigos de los niños.

---

<sup>5</sup> El aspecto de la página de administración puede verse en el Anexo 5. Manual de administración.

## Caso de uso: Añadir colegio

### Descripción breve:

Este caso de uso describe la funcionalidad del panel de administración que permite a un administrador añadir un colegio del proyecto *Camino seguro al cole* a la base de datos de la aplicación.

### Flujo de eventos:

#### Camino principal:

Este caso de uso comienza accediendo desde la página de administración <http://www.seguosalcole.comoj.com/administracion/login.php>.

En esta página debemos acceder con el usuario y la contraseña de administrador.

Una vez identificados accedemos al panel de administración desde el que tendremos que seleccionar la opción *Insertar nuevo colegio*, rellenando para ello los siguientes campos:

- Nombre del colegio
- Coordenadas en LatLng
- Dirección
- Código de seguridad para el registro
- Teléfono

#### Caminos secundarios:

No hay caminos secundarios en este caso de uso.

### Requisitos especiales:

No hay requisitos especiales en este caso de uso.

### Precondiciones:

El actor (administrador) debe estar registrado previamente como administrador, y recordar su usuario y contraseña.

El administrador debe haber seleccionado la opción *Insertar nuevo colegio* del panel de administración.

### Postcondiciones:

Queda insertado en la base de datos un nuevo colegio para la aplicación *Camino seguro al cole*.

## **Caso de uso: Obtener lista de colegios**

### **Descripción breve:**

Este caso de uso describe la funcionalidad del panel de administración que permite a un administrador obtener un fichero con la información de todos los colegios registrados en la aplicación *Camino seguro al cole*.

### **Flujo de eventos:**

#### **Camino principal:**

Este caso de uso comienza accediendo desde la página de administración <http://www.seguosalcole.comoj.com/administracion/login.php>.

En esta página debemos acceder con el usuario y la contraseña de administrador.

Una vez identificados accedemos al panel de administración desde el que tendremos que seleccionar la opción *Obtener un registro con todos los colegios almacenados*.

Tendremos que elegir dónde guardar el fichero, que será un archivo Excel que contendrá los siguientes campos:

- Nombre del colegio
- Dirección
- Coordenadas en (Lat, Lng)
- Número de comercios amigos de los niños asignados a este colegio
- Clave de seguridad para el registro de nuevos usuarios en la aplicación

#### **Caminos secundarios:**

No hay caminos secundarios en este caso de uso

### **Requisitos especiales:**

Para la correcta visualización del fichero deberá disponer de una versión de Microsoft Office Excel 2003 o superior.

### **Precondiciones:**

El actor (administrador) debe estar registrado previamente como administrador, y recordar su usuario y contraseña.

El administrador debe haber seleccionado la opción *Obtener un registro con todos los colegios almacenados* del panel de administración.

### **Postcondiciones:**

Queda en posesión del administrador un fichero con la información almacenada en la base de datos de todos los colegios registrados en la aplicación *Camino seguro al cole*.

## Caso de uso: Obtener rutas de un colegio

### Descripción breve:

Este caso de uso describe la funcionalidad del panel de administración que permite a un administrador obtener un fichero con las rutas subidas por los niños para un colegio determinado.

### Flujo de eventos:

#### Camino principal:

Este caso de uso comienza accediendo desde la página de administración <http://www.segurosalcole.comoj.com/administracion/login.php>.

En esta página debemos acceder con el usuario y la contraseña de administrador.

Una vez identificados accedemos al panel de administración desde el que tendremos que seleccionar la opción *Obtener un registro con todas las rutas pertenecientes a un colegio* insertando en el formulario el nombre del colegio del cual queremos obtener sus rutas subidas.

Tendremos que elegir dónde guardar el fichero<sup>6</sup>, que será un archivo Excel que contendrá los siguientes campos:

- Numero de usuario: identificador único de cada ruta.
- Ruta: lista de puntos LatLng que dan la ruta que sigue el usuario desde su casa hasta el colegio.
- Nombre del colegio: el nombre del colegio al que está asociado esa ruta.

#### Caminos secundarios:

No hay caminos secundarios en este caso de uso

#### Requisitos especiales:

Para la correcta visualización del fichero deberá disponer de una versión de Microsoft Office Excel 2003 o superior.

---

<sup>6</sup> El uso del fichero con las rutas puede verse en el Anexo 5. Manual de administración.

**Precondiciones:**

El actor (administrador) debe estar registrado previamente como administrador, y recordar su usuario y contraseña.

El administrador debe haber seleccionado la opción *Obtener un registro con todas las rutas pertenecientes a un colegio* del panel de administración.

El colegio del cual queremos obtener sus rutas debe ser un colegio registrado en la base de datos de la aplicación *Camino seguro al cole*.

**Postcondiciones:**

Queda en posesión del administrador un fichero con la información almacenada en la base de datos de las rutas subidas por los niños para un colegio determinado.

## **Caso de uso: Añadir administrador**

### **Descripción breve:**

Este caso de uso describe la funcionalidad del panel de administración que permite a un administrador añadir otro administrador al sistema.

### **Flujo de eventos:**

#### **Camino principal:**

Este caso de uso comienza accediendo desde la página de administración <http://www.segurosalcole.comoj.com/administracion/login.php>.

En esta página debemos acceder con el usuario y la contraseña de administrador.

Una vez identificados accedemos al panel de administración desde el que tendremos que seleccionar la opción *Añadir nuevo administrador* rellenando los siguientes campos del formulario:

- Número del usuario que se va a convertir en administrador
- Contraseña personal de acceso al panel de administración

#### **Caminos secundarios:**

No hay caminos secundarios en este caso de uso

### **Requisitos especiales:**

Para la correcta visualización del fichero deberá disponer de una versión de Microsoft Office Excel 2003 o superior.

### **Precondiciones:**

El actor (administrador) debe estar registrado previamente como administrador, y recordar su usuario y contraseña.

El administrador debe haber seleccionado la opción *Añadir nuevo administrador* del panel de administración.

### **Postcondiciones:**

Queda registrado un nuevo administrador en el sistema, permitiéndole el acceso a todas las funciones del panel de administración.

### 3.3.2. Vista lógica

#### Descripción

En esta sección se dividirá el sistema en subsistemas, y se expondrán las relaciones entre los paquetes y clases significativos de la arquitectura. Se mostrará un diagrama de paquetes y se hará una descripción de cada clase.

#### División en subsistemas

En la parte del cliente diferenciamos tres subsistemas, el de registro e identificación, el de gestión del camino seguro y el del servidor y la persistencia de datos. Interactúan entre ellos como se muestra en el siguiente diagrama.

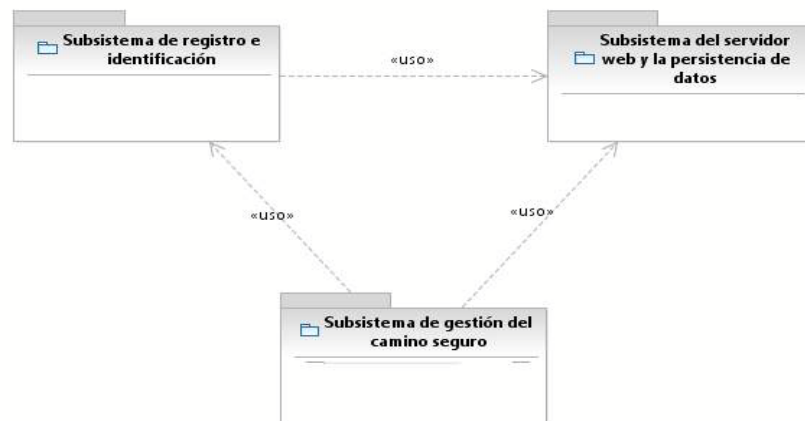


Figura 5: Diagrama de subsistemas.

**El subsistema de registro e identificación** se encarga de la seguridad en el acceso a la aplicación, no permitiendo el acceso a ninguna persona que no esté correctamente registrada e identificada con los mismos datos con los que realizó el registro. Se encarga de la realización de los casos de uso *Registrarse en la aplicación* e *Identificarse en la aplicación*.

**El subsistema de gestión del camino seguro** se encarga de dar soporte a las principales funcionalidades de la aplicación *Camino seguro al cole*, se encarga de la realización de los casos de uso *Consultar el camino seguro*, *Guiar al colegio*, *Valorar un comercio*, *Notificar una incidencia* y *Marcar mi camino seguro al cole*, apoyándose en el subsistema del servidor y persistencia de datos.

**El subsistema del servidor web y la persistencia de datos** se ocupa del almacenamiento y servicio de toda la información necesaria para el camino seguro al colegio, como las rutas seguras para cada colegio, los comercios amigos de los niños y el camino que cada usuario utiliza para ir al colegio. También se encarga de la realización de los casos de uso de administración.

## Paquetes y clases significativos de la arquitectura

### Diagrama de paquetes

En el siguiente diagrama se muestran todos los paquetes y clases que forman la aplicación *Camino seguro al cole*. Los paquetes *Misc*, *Camino* y *Server* son los subsistemas registro e identificación, gestión del camino seguro y servidor y persistencia de datos respectivamente.

El paquete *Intro* contiene una clase para la introducción de la aplicación. El paquete *Strings* contiene una clase con los avisos y mensajes que se le pasan al usuario y el paquete *Utils* contiene una clase *UtilGeneral* para decodificar las respuestas codificadas del servidor.

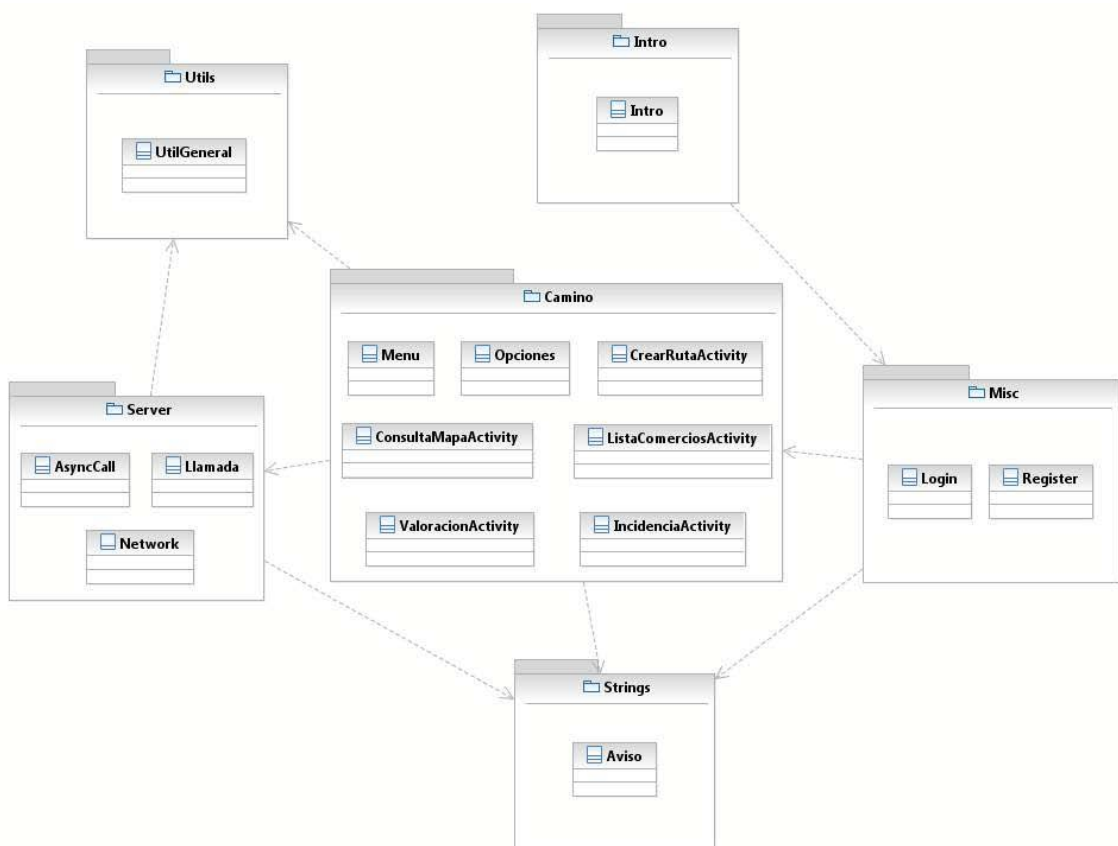


Figura 6: Diagrama de paquetes.



## Paquete Misc

El paquete *Misc* es el paquete encargado del registro y la identificación en la aplicación *Camino seguro al cole*, contiene las siguientes clases:

**Clase *Login*:** esta clase tiene una vista que nos permite ingresar los datos que hemos facilitado en el registro para acceder a las funcionalidades de la aplicación, así como iniciar un nuevo registro. Hace una llamada a la clase *UtilGeneral* para cifrar la contraseña en MD5 y proporcionar seguridad en el acceso. Cuando los datos están rellenos se envía una petición al servidor a través de las clases *AsyncCall* y *Llamada* y se decodifica la respuesta con la clase *UtilGeneral*. Si los datos son correctos se inicia el menú de la aplicación a través del cual podemos acceder a todas las funcionalidades.

**Clase *Register*:** esta clase tiene una vista que es un formulario de registro, habrá que rellenar correctamente todos los campos para poder identificarnos posteriormente con esos mismos datos. Cuando los datos están correctamente rellenos se envía una petición al servidor a través de las clases *AsyncCall* y *Llamada*.

## Paquete Server

El paquete *Server* es el encargado de la comunicación del cliente con el servidor web y la base de datos. Contiene las siguientes clases:

**Clase *AsyncCall*:** hace las peticiones al servidor web a través de la clase *Llamada* y ejecuta unas acciones antes y después de la ejecución de las llamadas. Tiene las peticiones para la identificación, el registro, la subida de la ruta y la valoración de un comercio y la obtención de la lista de comercios y la araña de movilidad para cada colegio.

**Clase *Llamada*:** ejecuta las peticiones al servidor y los accesos a la base de datos. Se usan las clases *HttpClient*, *HttpPost* y *HttpResponse*, contenidas en la librería *org.apache.http*. La respuesta obtenida del servidor se guarda en un atributo respuesta y se decodifica en los casos que sea necesario con la ayuda de la clase *UtilGeneral*.

## Paquete Camino

El paquete *Camino* es el encargado de implementar sobre sus clases las principales funcionalidades de la aplicación *Camino seguro al cole*. Contiene las siguientes clases:

**Clase *Menu*:** tiene una vista con un menú que nos permite acceder a cualquier funcionalidad de la aplicación y los lanzadores para cada actividad. La primera vez que iniciemos la aplicación nos mostrará un menú emergente para que activemos la localización GPS del dispositivo.

**Clase *Opciones*:** permite cambiar de usuario, seleccionar un nuevo colegio y mostrar una ayuda sobre el uso de la aplicación. Las dos primeras opciones surgen para la actualización de la araña de movilidad; la idea es que se lleve a los colegios la aplicación instalada y los niños de la clase suban sus rutas al cole pasándose el dispositivo entre todos.

**Clase *ConsultaMapaActivity*:** es la clase que soporta la funcionalidad de mostrar el camino seguro y guiar al camino seguro.

Obtiene la información de la base de datos a través de las clases *AsyncCall* y *Llamada* y la decodifica con la ayuda de la clase *UtilGeneral*. Dibuja toda la información obtenida (colegio, comercios y araña de movilidad) sobre un mapa de Google Maps v2.

Tiene los métodos necesarios para la geolocalización y para la actualización de la posición actual.

La opción mostrar cómo llegar a la ruta segura o un click largo sobre el mapa calculará el punto más cercano de acceso a la araña de movilidad según la posición actual del usuario y dibujará la ruta que proporciona Google Maps para realizar ese tramo a pie desde la posición actual del usuario.

**Clase *IncidenciaActivity*:** contiene la dirección de enlace con la sección de gestiones y trámites de la página web del Ayuntamiento de Madrid. Se encarga de la realización del caso de uso *Notificar una incidencia*.

**Clase *ListaComerciosActivity*:** se encarga de la realización del caso de uso *Valorar un comercio*, junto con la clase *ValoracionActivity*.

Permite mostrar una lista de comercios amigos de los niños para el colegio seleccionado por el usuario, junto con su valoración media.

Obtiene la información de la base de datos mediante las clases *AsyncCall* y *Llamada*, y decodifica la respuesta del servidor con la clase *UtilGeneral*.

Con un click sobre el establecimiento de la lista que queramos lanzamos una nueva actividad para darle una valoración.

**Clase *ValoracionActivity*:** permite enviar una valoración de un comercio previamente seleccionado. El usuario podrá hacer una valoración de una a cinco estrellas (en intervalos de media estrella) del comercio que quiera, y ver la valoración media actual. Al pulsar sobre el botón *Enviar valoración* se lanza una petición al servidor mediante el método habitual y se actualiza el contenido de la base de datos.

**Clase *CrearRutaActivity*:** es la clase que se encarga de la realización del caso de uso *Marcar mi camino seguro*, que permite subir a la base de datos el camino que sigue cada uno de los usuarios desde su casa hasta el colegio. Tiene todos los métodos necesarios para la geolocalización del usuario, cogiendo como punto de partida en la ruta la localización actual. Haciendo click en cualquier posición del mapa se recoge su latitud y su longitud y se añade a una estructura para seguir la ruta, además de dibujar una línea entre los dos puntos anteriores.

Con el botón de opciones podemos deshacer el último tramo si nos hemos equivocado o empezar de nuevo, así como enviar la ruta actual a la base de datos haciendo uso de las clases destinadas a la conexión *AsyncCall* y *Llamada*.

### 3.3.3. Vista de procesos

En esta vista se describe los procesos e hilos que ejecuta el sistema. En el cliente Android se ejecuta la mayoría del tiempo un solo hilo, salvo cuando se hacen peticiones al servidor a través de la clase *AsyncCall* con el método *execute*, que crea un nuevo hilo para ejecutar sus acciones en segundo plano (*doInBackground*). Una vez que termina de ejecutar sus acciones cierra el hilo y continúa con la ejecución del método *onPostExecute* sobre el hilo principal. Lo mismo ocurre con la localización GPS, que crea un hilo para la actualización de la posición actual del usuario.

### 3.3.4. Vista de despliegue

#### Elementos de la distribución

Los elementos de la distribución de la aplicación *Camino seguro al cole* son tres, y se corresponden con cada uno de los nodos que posee el sistema (cliente Android, servidor web y base de datos).

**Aplicación para cliente Android:** es un archivo llamado *Camino.apk*, que debe ser instalado en el dispositivo según se especifica en el apartado Instalación del Manual de Usuario. Se comunica con el servidor web mediante interacción *HTTP*.

**Servidor web:** es un servidor *Apache HTTP* con páginas *PHP* que hacen las transferencias entre la base de datos y el cliente mediante consultas *SQL*. Las respuestas se obtienen en formato *JSON* y se decodifican con la ayuda de la clase *UtilGeneral* en el cliente.

**Base de datos:** se encarga del almacenamiento persistente de la información necesaria para el cliente. Recibe las consultas del servidor web mediante el uso de librerías *MySQL* de *PHP*<sup>7</sup>. La organización de las tablas que contiene la base de datos está especificada en el apartado Vista de datos.

---

<sup>7</sup> En los Anexo 3 se encuentra el código de todas las páginas *PHP*.

## Despliegue

La arquitectura del sistema está dividida en tres nodos para dar más escalabilidad al sistema, aunque en el momento de desplegarlo para las pruebas, los nodos de servidor web y base de datos fueron desplegados sobre la misma dirección física, como se ve en el siguiente diagrama de despliegue.

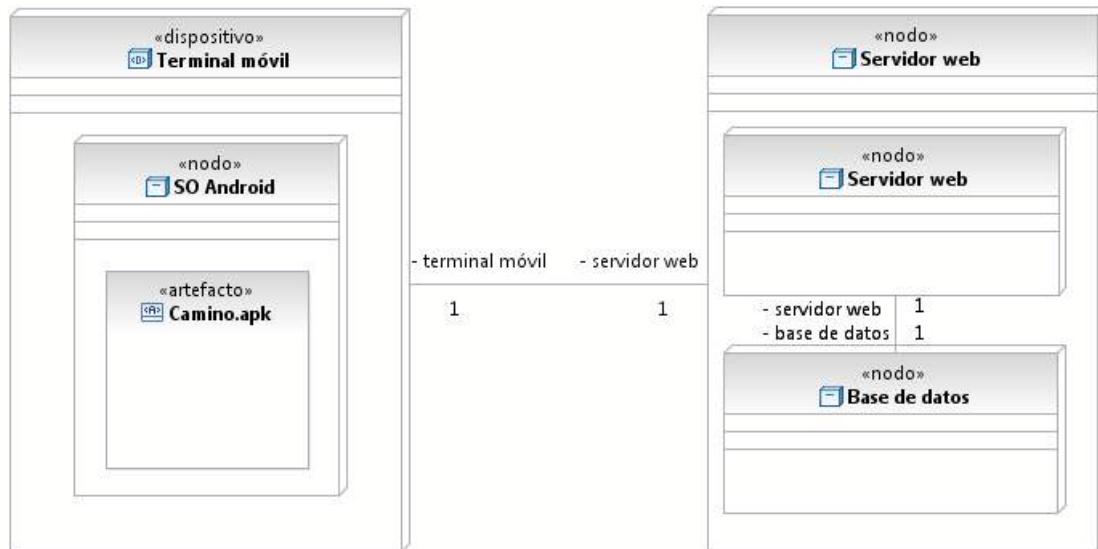


Figura 7: Diagrama de despliegue.

### 3.3.5. Vista de implementación

En esta vista veremos cómo se han implementado cada uno de los subsistemas lógicos en los que está dividida la aplicación *Camino seguro al cole*. Podemos ver la división en subsistemas en el diagrama de la Figura 5.

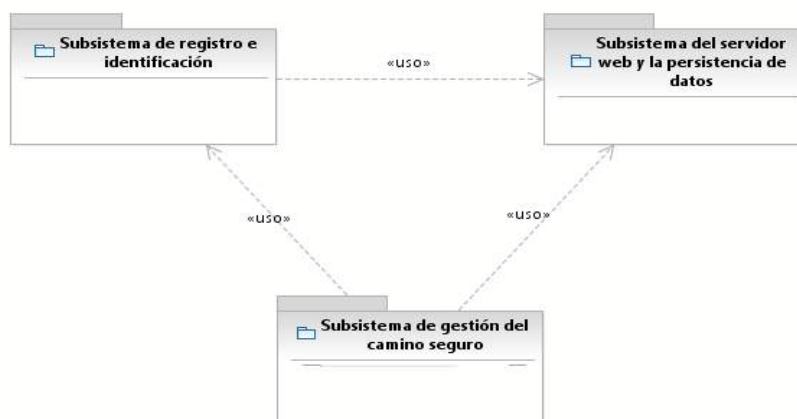


Figura 5: Diagrama de subsistemas.

## Subsistema de registro e identificación

En la implementación, a este subsistema se le ha dado el nombre *Misc*, y es el paquete que se encarga del registro e identificación de un usuario para permitirle el acceso al resto de funcionalidades. Consta de dos clases, *Login* y *Register*, como vimos en el apartado de vista lógica. A continuación mostraremos un diagrama de clases con los métodos de cada clase y una explicación del funcionamiento de los métodos más importantes.

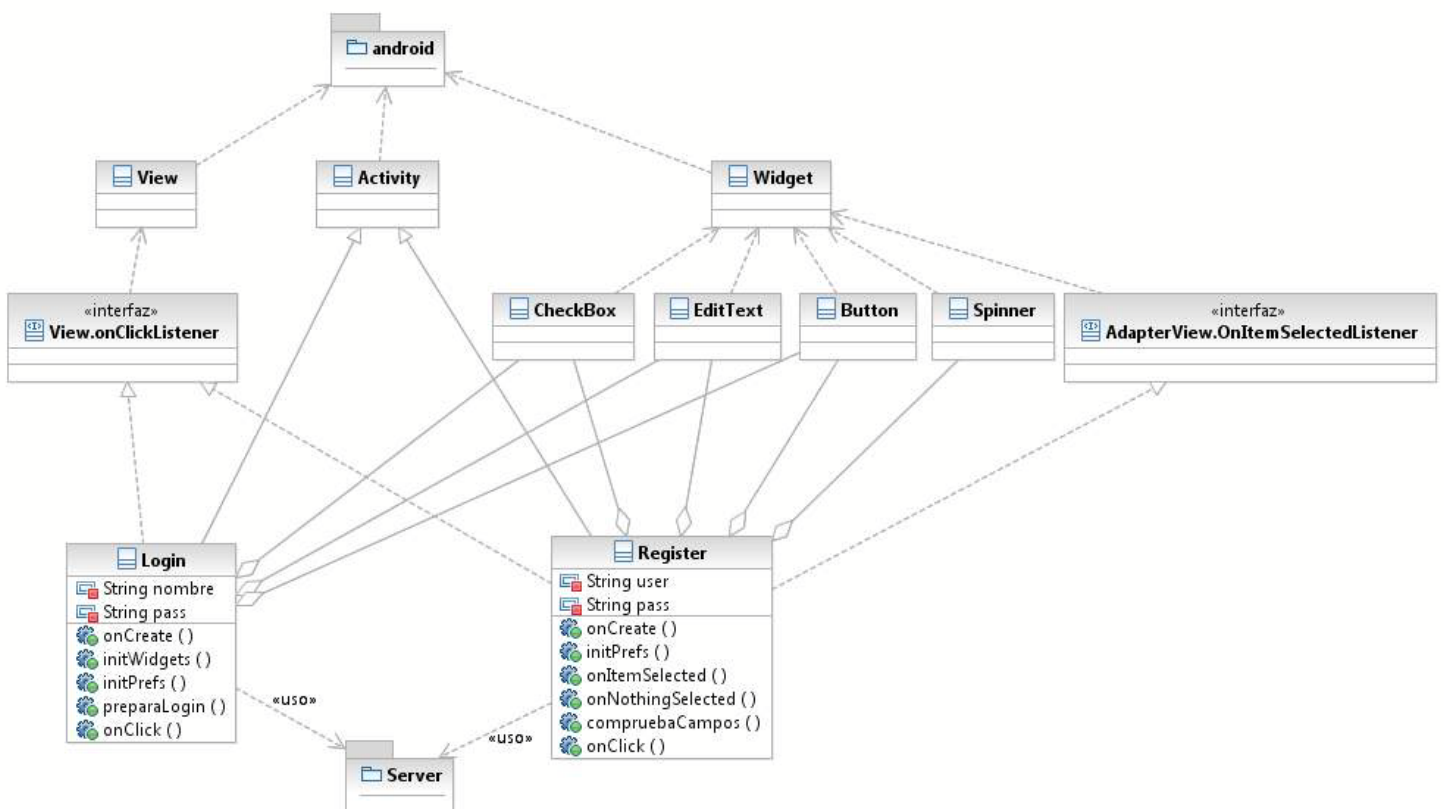


Figura 8: Diagrama de clases subsistema de registro e identificación.

- **Clase *Login*:**
  - Método ***onCreate()***: asigna la vista del *Login*, llama a los métodos *initWidgets()* e *initPrefs()* y asigna el método *onClick()* a los botones *Registro*, *Login* y *Olvidaste tu contraseña*.
  - Método ***initWidgets()***: inicializa los cuadros de texto, los botones y las cajas de marcado.
  - Método ***initPrefs()***: inicializa las preferencias, obteniendo la configuración global de las opciones.
  - Método ***preparaLogin()***: codifica la contraseña con el algoritmo de reducción criptográfico MD5. Devuelve *verdadero* si se ha preparado correctamente y *falso* en caso contrario.
  - Método ***onClick()***: asocia las acciones a los botones *Login* , *Registro* y *Olvidaste tu contraseña*. Si pulsamos sobre el botón *Acceder* hacemos la petición al servidor a través de los métodos de las clases *AsyncCall* y *Llamada* (consultar el subsistema del servidor web y la persistencia de datos dentro de esta misma sección). Si pulsamos sobre el botón *Registro* iniciamos una nueva actividad para registrarnos en el sistema. Si pulsamos sobre el botón *Olvidaste tu contraseña* se nos enviará un email a la dirección proporcionada en el registro con una nueva contraseña, que anulará la antigua.

- **Clase *Register*:**
  - Método ***onCreate()***: asigna la vista del registro e inicializa los componentes de la vista (cuadros de texto, botones, cajas de marcado y el *Spinner* para la selección del colegio).
  - Método ***initPrefs()***: inicializa las preferencias.
  - Métodos ***onItemSelected()*** y ***onNothingSelected()***: métodos asociados a la selección del colegio dentro del *Spinner*.
  - Método ***compruebaCampos()***: comprueba que los campos que ha rellenado el usuario cumplan con los requisitos de corrección. Devuelve *verdadero* si cumplen los requisitos y *falso* en caso contrario, junto con el aviso correspondiente para que el usuario pueda corregir la información.
  - Método ***onClick()***: asigna las acciones a los botones *Aceptar las condiciones de uso* y *Confirmar*.  
Si se pulsa el botón *Confirmar* se lanza una petición al servidor a través de las clases *AsyncCall* y *Llamada* para el registro de un nuevo usuario en el sistema.

## Subsistema de gestión del camino seguro

En la implementación, a este subsistema se le ha dado el nombre *Camino*, y es el paquete que se encarga de implementar sobre sus clases las principales funcionalidades de la aplicación (*Consultar el camino seguro*, *Guiar al colegio*, *Notificar una incidencia*, *Valorar un comercio* y *Marcar mi camino seguro*). A continuación mostraremos un diagrama de clases con los métodos más importantes de cada clase y una explicación de su funcionamiento.



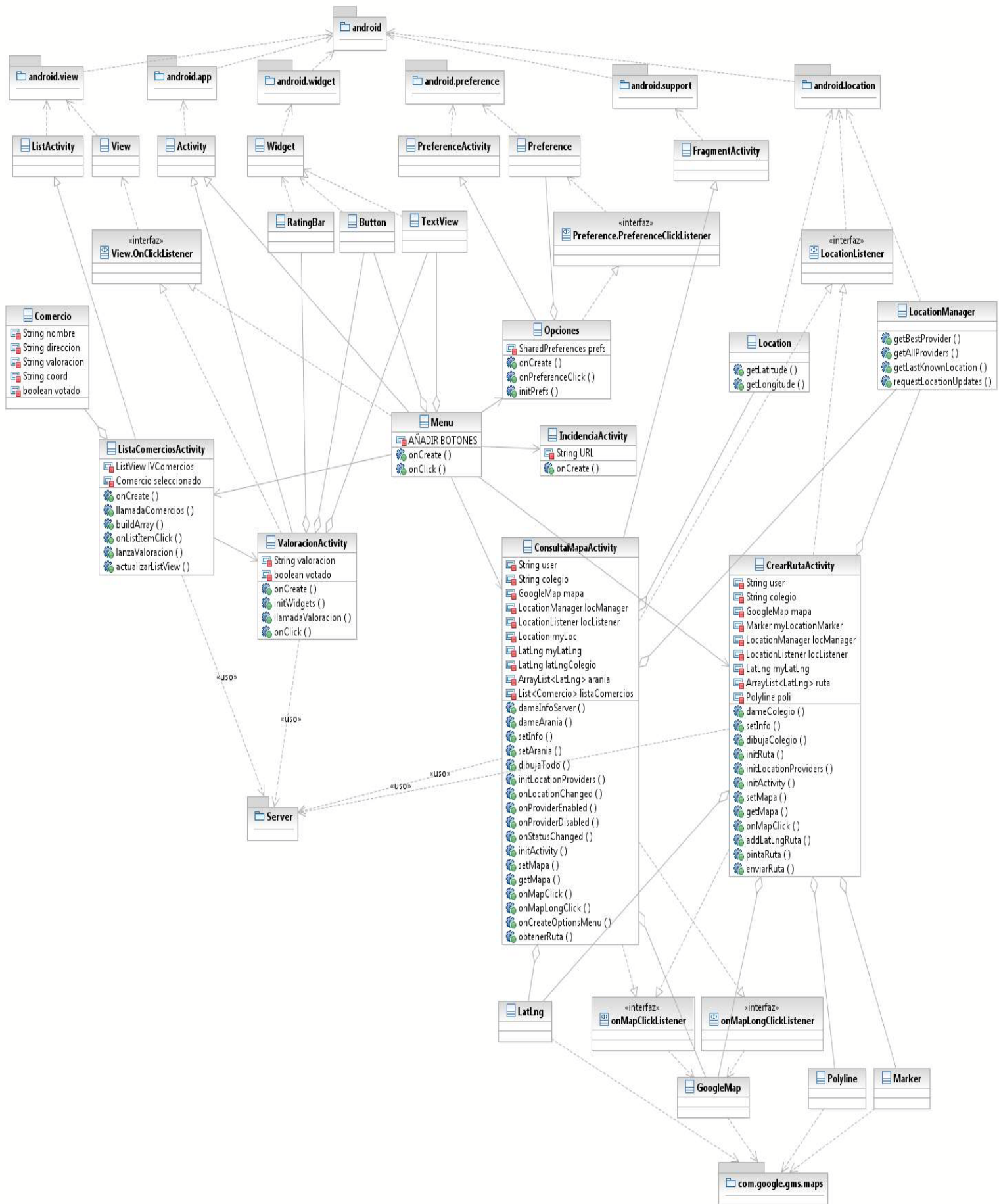


Figura 9: Diagrama de clases subsistema de gestión del camino seguro.

- **Clase *Comercio*:**
  - Métodos ***getNombre()***, ***getDireccion()***, ***getValoracion()*** y ***getCoord()***: son los métodos para obtener la información de los atributos de la clase como el nombre, la dirección, la valoración media y las coordenadas de un comercio. Devuelven un *String* con la información.
  - Métodos ***setNombre()***, ***setDireccion()***, ***setValoracion()*** y ***setCoord()***: son los métodos para asignar valor a los atributos de la clase como el nombre, la dirección, la valoración media y las coordenadas de un comercio. Reciben como parámetro un *String* con el valor que se desea asignar.
- **Clase *Menu*:**
  - Método ***OnCreate()***: inicializa los botones y les asigna un *onClickListener* para que puedan ejecutar acciones cuando se pulse sobre ellos. En el método *onClick()* se lanzarán las diferentes actividades.
  - Método ***onClick()***: recibe la vista como parámetro para saber qué botón es el que ha sido pulsado y según el que haya sido lanza la actividad que corresponda.
- **Clase *ConsultaMapaActivity*:**
  - Método ***onCreate()***: asigna la vista del mapa de Google Maps v2, inicia la actividad del mapa, crea las estructuras para almacenar la información de la base de datos, inicializa la búsqueda de proveedores de localización y hace la llamada al servidor con el método *dameInfoServer()* que se explicará más abajo.
  - Método ***initPrefs()***: obtiene el colegio seleccionado en el registro de las opciones.
  - Métodos ***dameInfoServer()*** y ***dameArania()***: realizan la llamada al servidor a través de las clases *AsyncCall* y *Llamada* para la obtención de la posición del colegio, la araña de movilidad y la posición de los comercios amigos de los niños para el colegio seleccionado. En el método *onPostExecute()* de la clase *AsyncCall* se llama a los métodos *setInfo()* y *setArania()*. Se utiliza el único método de la clase *Network* para saber si la red está disponible antes de ejecutar la llamada al servidor.

- Métodos ***setInfo()*** y ***setArania()***: usan la clase *UtilGeneral* para decodificar la respuesta del servidor en formato *JSON*. Llamam a los métodos *decodAll()* y *decodArania()* respectivamente.
- Método ***dibujaTodo()***: llama a los métodos *dibujaComercios()*, *dibujaArania()* y *dibujaColegio()* para dibujar sobre el mapa los comercios, la araña de movilidad y el colegio.
- Método ***initLocationProviders()***: inicia la búsqueda del mejor proveedor de servicios de localización y ajusta la frecuencia de actualización de la posición que se actualizará con el método *actualizarPosición()*.
- Métodos ***onLocationChanged()***, ***onProviderDisabled()***, ***onProviderEnabled()*** y ***onStatusChanged()***: son métodos relacionados con la localización del usuario, dan información sobre el proveedor de servicio y actualizan la posición del usuario cuando cambia su localización.
- Método ***initActivity()***: inicia la actividad del mapa de Google v2, y asigna un *onClickListener()* y un *onLongClickListener()* para ejecutar acciones al pulsar o al mantener pulsado un punto del mapa.
- Método ***onMapLongClick()***: llama al método *obtenerRuta()* para calcular la ruta a pie desde la posición actual del usuario hasta el punto más cercano de la araña de movilidad calculado mediante el método *minLatLng()*. Decodifica la respuesta del servidor con el método *decodeJSON()*, que a su vez llama al método *getListaLatLng()* para separar el *String* obtenido en puntos *LatLng* y dibuja la ruta dibujando tramo a tramo cada uno de los puntos en el método *dibujaRuta()*.
- Método ***onCreateOptionsMenu()***: asigna como *MenuInflater* un menú con un botón para calcular la ruta a pie desde la posición actual hasta el punto más cercano de la araña de movilidad, igual que se hacía con el método *onMapLongClick()*, ya que es poco intuitivo. También muestra las opciones de moverse libremente sobre el mapa y fijar la vista del mapa en la posición actual del usuario.  
El menú se despliega pulsando la tecla de opciones del dispositivo y si pulsamos sobre el botón *Mostrar ruta segura* se llama al método *minLatLng()* para calcular el punto de entrada a la araña de movilidad y se llama al método *obtenerRuta()* con la posición actual como origen y el punto de entrada a la araña de movilidad como destino.

- Método ***obtenerRuta()***: es el método que realiza la llamada *HTTP* sobre la página de Google Maps para la obtención de la ruta a pie desde el origen hasta el destino pasados como parámetros. Se obtiene en formato *JSON* y habrá que decodificarlo con el método *decodeJSON()*.
- Métodos ***crearDialogoInet()***, ***crearDialogoActivaGps()***, ***crearDialogoConfirmaReintento()***: son métodos de interacción con el usuario, sirven para informar de que el acceso a Internet o la localización GPS no están disponibles.
- Clase ***ListaComerciosActivity***:
  - Método ***onCreate()***: crea las estructuras para almacenar la lista de comercios, obtiene el colegio seleccionado por el usuario de opciones con el método *initPrefs()* y llama al método *llamadaComercios()*.
  - Método ***llamadaComercios()***: es el método en el que se hace la petición al servidor a través de las clases *AsyncCall* y *Llamada* (método *getComercios()*). En el método *onPostExecute* de la clase *AsyncCall*, una vez recibida la respuesta del servidor se llama al método *setEstabs()*, que a su vez llama al método *buildArray()* para la construcción de la lista de comercios amigos de los niños asignados al colegio que el usuario seleccionó en el registro.
  - Método ***buildArray()***: decodifica la respuesta en *JSON* del servidor con la ayuda del método *decodComercios()* de la clase *UtilGeneral* y crea una lista de establecimientos con el método *setListaAdaptada()*.
  - Método ***onListItemClick()***: lanza el método *lanzaValoracion()* sobre el comercio seleccionado de la lista.
  - Método ***lanzaValoracion()***: inicia la actividad para darle una valoración a un comercio pasándole la información necesaria para garantizar que sólo hay una valoración de cada comercio por usuario.

- **Clase *ValoracionActivity*:**
  - Método ***onCreate()***: asigna la vista de valoración de un comercio e inicializa los cuadros de texto, los botones y la barra de valoraciones mediante el método *initWidgets()*.
  - Método ***onClick()***: recibe la vista como parámetro para saber qué botón se ha pulsado, y cuando se pulsa el botón *Enviar valoración* se llama al método *llamadaValoracion()*.
  - Método ***llamadaValoracion()***: es el método que ejecuta la petición al servidor a través de las clases *AsyncCall* y *Llamada* (método *subirValoracion()*). Se mostrará un mensaje confirmando la subida correcta o un mensaje avisando de que no se ha podido realizar. Al subir la valoración, se recalculará la valoración media del establecimiento.
  
- **Clase *crearRutaActivity*:**
  - Método ***onCreate()***: asigna la vista del mapa de Google Maps v2 e inicia la actividad del mapa con el método *initActivity()*. Carga los comercios e inicializa las preferencias y el colegio con el método *initPrefs()*. Inicia la búsqueda de proveedores de localización con el método *initLocationProviders()* y añade la posición actual como primer punto de partida en la ruta si la localización está disponible. Por último hace la petición al servidor a través del método *dameColegio()*, detallado más abajo.
  - Método ***initPrefs()***: Obtiene el colegio y el usuario de las opciones.
  - Método ***initLocationProviders()***: Igual que el método con el mismo nombre de la clase *ConsultaMapaActivity*. Fija unos criterios que debe cumplir el proveedor de servicios de localización y se queda con el mejor posible. Ajusta la frecuencia de actualización.
  - Método ***mostrarPosicion()***: es un método que permite mostrar con un marcador la posición actual del usuario sobre el mapa. También fija el intervalo de actualización del marcador.
  - Métodos ***initActivity()***, ***onLocationChanged()***, ***onProviderDisabled()***, ***onProviderEnabled()*** y ***onStatusChanged()***: son métodos asociados a los servicios de localización, tienen la misma función que sus homónimos de la clase *ConsultaMapaActivity*.

- Método **dameColegio()**: realiza la petición al servidor a través de las clases *AsyncCall* y *Llamada* (método *getColegio()*). En el método *onPostExecute* de la clase *AsyncCall* se llama al método *setInfo()* con la respuesta del servidor.
- Método **setInfo()**: decodifica la respuesta del servidor con la ayuda del método *decodColegio()* de la clase *UtilGeneral*. Si se decodifica correctamente se llama a la función *dibujaColegio()* que pondrá un marcador en el mapa en la posición del colegio.
- Método **cargaComercios()**: carga los comercios asociados al colegio seleccionado y los añade como marcadores en el mapa con el método *addItems()*.
- Método **initRuta()**: inicializa la ruta con la posición actual. La ruta es una estructura que se gestiona como una pila.
- Método **onMapClick()**: añade la posición en *LatLng* del punto en el que pulsemos en el mapa a la estructura ruta mediante el método *addLatLngRuta()* y pinta la ruta que haya almacenada en la estructura hasta el momento con el método *pintaRuta()*, trazando líneas entre los puntos de la estructura.
- Método **onCreateOptionsMenu()**: asigna como *MenuInflater* un menú con tres botones, *Deshacer último tramo*, *Empezar de nuevo* y *Enviar mi ruta*. El *MenuInflater* se despliega pulsando la tecla de opciones del dispositivo.
- Método **onOptionsItemSelected()**: se encarga de las acciones al pulsar sobre los botones definidos en el método anterior. Como se ha mencionado anteriormente, la gestión de la estructura ruta se realiza como si fuera una pila, por tanto, para deshacer el último tramo eliminamos de la estructura la cima o el último elemento introducido.  
Si pulsamos sobre el botón *Empezar de nuevo* reiniciamos la estructura ruta llamando al método *initRuta()* y si pulsamos sobre el botón *Enviar mi ruta* se llama al método *enviarRuta()* explicado a continuación.
- Método **enviarRuta()**: introduce la información almacenada en la estructura ruta en un *String* y hace la llamada al servidor mediante las clases *AsyncCall* y *Llamada* (método *subirRuta()*). Muestra un aviso para saber si la subida se ha podido completar correctamente o no.

## Subsistema del servidor web y la persistencia de datos

En la implementación, a este subsistema se le ha dado el nombre *Server*, y es el paquete que se encarga de las llamadas al servidor web, tanto para almacenamiento persistente en la base de datos como para la recuperación de dichos datos. Consta de dos clases, *AsyncCall* y *Llamada*, como vimos en el apartado de vista lógica. El código de todas las páginas *PHP* referenciadas desde la clase *Llamada* se puede ver en los anexos. A continuación mostraremos un diagrama de clases con los métodos de cada clase y una explicación del funcionamiento de los métodos más importantes.

Figura 10: Diagrama de clases subsistema del servidor web y la persistencia de datos.



- **Clase *AsyncCall*:**

- Método ***AsyncCall()***: es el constructor, asigna la actividad que se pasa como parámetro como actividad padre, crea un objeto de la clase *Llamada* y un diálogo de progreso.
- Método ***onPreExecute()***: son las acciones que se realizan antes de ejecutar la llamada al servidor, en nuestro caso se muestra un diálogo de progreso.
- Método ***doInBackground()***: son las acciones que se ejecutan con la llamada a *execute*, se ejecutan sobre un nuevo hilo que se crea en segundo plano.  
Tiene asignadas las acciones para las tareas de identificación, registro, obtención de la araña de movilidad, obtención de la lista de comercios, subida de la ruta seguida por un usuario, subida de la valoración de un comercio y obtención conjunta de la posición del colegio, comercios y araña de movilidad.  
Dentro de cada tarea, llama al método de la clase *Llamada* que se encarga de realizar las peticiones mediante *HttpPost* y *HttpResponse* y obtiene la respuesta dada por el servidor en formato *JSON*, que se decodificará en los casos que sea necesario con ayuda de la clase *UtilGeneral*.
- Método ***onPostExecute()***: son las acciones que se realizan después de ejecutar la llamada *execute*. Muestra los avisos correspondientes según la respuesta del servidor y lanza las actividades o métodos que correspondan con la información obtenida. Hay acciones para cada una de las tareas especificadas en el método *doInBackground()*.

- **Clase *Llamada*:**

- Método ***Llamada()***: es el constructor de la clase, guarda el contexto de la actividad que le llama e inicializa la respuesta a vacío.
- Método ***getRespuesta()***: devuelve en un *String* la respuesta del servidor.
- Método ***login()***: es el método en el que se hace la petición de consulta de la base de datos desde el servidor mediante la llamada con *HttpPost* a la página *login.php*. Devuelve *verdadero* si se realizó correctamente y *falso* en caso contrario.
- Método ***register()***: es el método en el que se hace la petición al servidor para el registro de un nuevo usuario. Hace un *HttpPost* con la página *register.php*. Devuelve *verdadero* si se realizó correctamente y *falso* en caso contrario.



- Método **getAll()**: este método obtiene de la base de datos toda la información necesaria para la funcionalidad de consulta del camino seguro (colegio, araña de movilidad y comercios asignados al colegio). Realiza una petición al servidor mediante el *HttpPost* a la página *getAll.php*. Reemplaza los caracteres especiales devueltos en la respuesta del servidor con la ayuda de la clase *UtilGeneral*. Devuelve *verdadero* si se realizó correctamente y *falso* en caso contrario.
- Método **spider()**: este método obtiene de la base de datos la araña de movilidad para el colegio seleccionado en el registro. Realiza una petición al servidor mediante el *HttpPost* a la página *getArania.php*. Devuelve *verdadero* si se realizó correctamente y *falso* en caso contrario.
- Método **getComercios()**: este método obtiene de la base de datos la información de los comercios asignados al colegio seleccionado en el registro. Realiza una petición al servidor mediante el *HttpPost* a la página *getComercios.php*. Reemplaza los caracteres especiales devueltos en la respuesta del servidor con la ayuda de la clase *UtilGeneral*. Devuelve *verdadero* si se realizó correctamente y *falso* en caso contrario.
- Método **subirRuta()**: es el método en el que se hace la petición al servidor para la subida de la ruta que un usuario sigue desde su casa hasta el colegio para la actualización de la araña de movilidad. Hace un *HttpPost* con la página *setRuta.php*. Devuelve *verdadero* si se realizó correctamente y *falso* en caso contrario.
- Método **subirValoracion()**: es el método que realiza la petición al servidor para la subida de una nueva valoración de un comercio amigo de los niños. Realiza una petición al servidor mediante el *HttpPost* a la página *setValoracion.php*. Devuelve *verdadero* si se realizó correctamente y *falso* en caso contrario.
- Método **getColegio()**: es el método que obtiene de la base de datos la información de un colegio dado (número de teléfono, dirección, número de establecimientos asociados y coordenadas). Realiza una petición al servidor mediante el *HttpPost* a la página *getColegio.php*. Reemplaza los caracteres especiales devueltos en la respuesta del servidor con la ayuda de la clase *UtilGeneral*. Devuelve *verdadero* si se realizó correctamente y *falso* en caso contrario.

### 3.3.6. Vista de datos

En esta vista se muestra la estructura de almacenamiento de la información persistente. En la base de datos se almacena toda la información necesaria para la gestión del camino seguro. Consta de siete tablas, cuyos campos se pueden ver más abajo, así como una breve descripción de la función de cada una.

Hay una tabla para almacenar toda la información de los usuarios proporcionada en el registro y otra para almacenar toda la información de los colegios inscritos en el programa, así como una tabla para almacenar la información de los establecimientos amigos de los niños. Hay una tabla para guardar las valoraciones que los usuarios pueden hacer de un establecimiento amigo de los niños. También hay una tabla *Adheridos*, que relaciona los establecimientos asociados a cada colegio y las coordenadas del establecimiento.

La tabla *Arañas* almacena por tramos (origen-destino) la araña de movilidad para cada colegio. Por último hay una tabla para almacenar la información de la ruta que cada niño sigue desde su casa al colegio proporcionada por la funcionalidad *Marcar mi camino seguro*.

Estructura de la tabla ***Adheridos***

Campo	Tipo	Null	Por defecto
nombCol	varchar(50)	Yes	NULL
nombEst	varchar(50)	Yes	NULL
coordEst	varchar(50)	Yes	NULL

Estructura de la tabla ***Arañas***

Campo	Tipo	Null	Por defecto
nombCol	varchar(50)	Yes	NULL
origen	varchar(50)	Yes	NULL
destino	varchar(50)	Yes	NULL

Estructura de la tabla ***Colegios***

Campo	Tipo	Null	Por defecto
nombCol	varchar(50)	Yes	NULL
dir	varchar(100)	Yes	NULL
numUs	int(9)	Yes	NULL
coord	varchar(50)	Yes	NULL
numEst	int(11)	Yes	NULL
codigo	Int(11)	Yes	NULL
clave	int(11)	Yes	NULL

Estructura de la tabla ***Establecimientos***

<b>Campo</b>	<b>Tipo</b>	<b>Null</b>	<b>Por defecto</b>
nombEst	varchar(50)	Yes	NULL
dir	varchar(100)	Yes	NULL
Coord.	varchar(50)	Yes	NULL
nombCol	varchar(50)	Yes	NULL
valoracion	tinyint(2)	Yes	NULL

Estructura de la tabla ***Ruta***

<b>Campo</b>	<b>Tipo</b>	<b>Null</b>	<b>Por defecto</b>
numUs	int(9)	Yes	NULL
ruta	varchar(50)	Yes	NULL
nombCol	varchar(50)	Yes	NULL

Estructura de la tabla ***Usuarios***

<b>Campo</b>	<b>Tipo</b>	<b>Null</b>	<b>Por defecto</b>
numUs	int(9)	Yes	NULL
nombre	varchar(50)	Yes	NULL
dir	varchar(100)	Yes	NULL
tutor	varchar(50)	Yes	NULL
tlfTutor	int(9)	Yes	NULL
nombCol	varchar(50)	Yes	NULL
pass	varchar(32)	Yes	NULL
tkn	varchar(6)	Yes	NULL

Estructura de la tabla ***Valoraciones***

<b>Campo</b>	<b>Tipo</b>	<b>Null</b>	<b>Por defecto</b>
nombEst	varchar(50)	Yes	NULL
dir	varchar(50)	Yes	NULL
numUs	int(9)	Yes	NULL
puntuacion	tinyint(2)	Yes	NULL

## Capítulo 4. Técnicas especiales utilizadas

### 4.1. Fórmula de Haversine

El caso de uso *Guiar al camino seguro* describe la funcionalidad de la aplicación de guiar al usuario desde el punto en el que se encuentre actualmente hasta el punto más cercano de la araña de movilidad. Para el cálculo del punto más cercano al usuario utilizamos la fórmula de Haversine<sup>[7]</sup>.

La fórmula de Haversine es una importante ecuación para la navegación astronómica, en cuanto al cálculo de la distancia entre dos puntos de un globo sabiendo su longitud y su latitud. Es un caso especial de la ley de los semiversenos, que relaciona los lados y los ángulos de un triángulo dentro de una esfera.

El nombre se deriva del hecho de que suele expresarse en términos de la función haversine, dada por  $\text{haversin}(\vartheta) = \sin^2(\vartheta/2)$ .

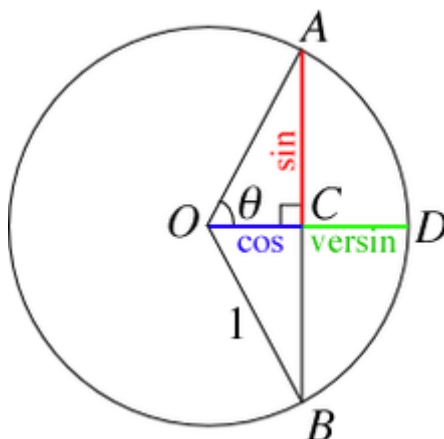


Figura 11: seno, coseno y verseno de  $\vartheta$  en base a la circunferencia goniométrica.

Para cualquier par de puntos sobre una esfera:

$$\text{haversin}\left(\frac{d}{r}\right) = \text{haversin}(\phi_2 - \phi_1) + \cos(\phi_1) \cos(\phi_2) \text{haversin}(\lambda_2 - \lambda_1)$$

Donde:

haversin es la función haversine,  $\text{haversin}(\vartheta) = \sin^2(\vartheta/2) = (1 - \cos(\vartheta))/2$ .

$d$  es la distancia esférica entre los dos puntos.

$r$  es el radio de la esfera (en nuestro caso el radio de la Tierra).

$\phi_1$  es la latitud del primer punto.

$\phi_2$  es la latitud del segundo punto.

$\lambda_1$  es la longitud del primer punto.

$\lambda_2$  es la longitud del segundo punto.

Ya sólo tenemos que despejar  $d$  de la ecuación:

$$d = 2r \arcsin \left( \sqrt{\text{havversin}(\phi_2 - \phi_1) + \cos(\phi_1) \cos(\phi_2) \text{havversin}(\lambda_2 - \lambda_1)} \right)$$

$$= 2r \arcsin \left( \sqrt{\sin^2 \left( \frac{\phi_2 - \phi_1}{2} \right) + \cos(\phi_1) \cos(\phi_2) \sin^2 \left( \frac{\lambda_2 - \lambda_1}{2} \right)} \right)$$

En esta función el argumento a la función haversine debe darse en radianes. Cuando se aplica sobre la Tierra es una aproximación, ya que no es una esfera perfecta, pero es más que suficiente para resolver el problema que nos ocupa.

Vemos ahora la implementación de la fórmula de Haversine para calcular la distancia entre los puntos “uno” y “dos”. A este método se le llama desde el método *minLatLng()*, que devuelve el punto de la araña de movilidad más cercano a la posición actual del usuario.

Se aplica para todos los puntos de entrada a la araña de movilidad, siendo “uno” la posición actual del usuario y “dos” cada uno de los puntos de la araña de movilidad.

```
private LatLng minLatLng() {
    LatLng min = null;
    if (this.arania != null) {
        int i = 0;
        LatLng actual = null;
        double distancia_min = Double.MAX_VALUE;
        while (i < arania.size() - 1) {
            actual = arania.get(i);
            double aux = HaverSineDistance(myLatLng, actual);
            if (distancia_min > aux){
                min = actual;
                distancia_min = aux;
            }
            i++;
        }
    }
    return min;
}
```

```

public static double HaverSineDistance(LatLng uno, LatLng dos) {

    double lat1,lng1,lat2,lng2;

    lat1 = Math.toRadians(uno.latitude);
    lng1 = Math.toRadians(uno.longitude);
    lat2 = Math.toRadians(dos.latitude);
    lng2 = Math.toRadians(dos.longitude);

    double dlon = lng2 - lng1;
    double dlat = lat2 - lat1;

    double a = Math.sin(dLat/2) * Math.sin(dLat/2) +
    Math.cos(Math.toRadians(lat1)) *
    Math.cos(Math.toRadians(lat2)) * Math.sin(dLon/2) * Math.sin(dLon/2);

    double c = 2 * Math.asin(Math.sqrt(a));

    return EARTH_RADIUS * c;
}

```



## Capítulo 5. Conclusiones y trabajo futuro

### 5.1. Conclusiones

Durante el desarrollo de la aplicación *Camino seguro al cole* la especificación no ha variado demasiado, cambiando sólo pequeños detalles para ajustarnos a los requisitos. Esto es debido, en parte, a que el proyecto “Madrid a pie: Camino seguro al cole” es un proyecto consolidado.

Al principio del desarrollo tuvimos que hacer un cambio en la visión del proyecto, ya que según el Instituto Nacional de Estadística, la edad media para que un niño tuviera su primer móvil se situaba en torno a los 13 años, y este móvil no suele ser un móvil que cumpla con los requisitos necesarios para el funcionamiento de la aplicación. Por eso se planteó la aplicación como una herramienta familiar, para que los padres la instalaran en su dispositivo, y padres e hijos pudieran ganar juntos en seguridad y autonomía. Sin embargo, la tendencia actual es a adelantar la edad de inicio del primer móvil, situándose en torno a los 12 años, por lo que en un futuro quizás tenga sentido hablar de una aplicación destinada al uso exclusivo de los niños.

Tuvimos también que realizar cambios en el envío de notificaciones, ya que la API de la aplicación WhatsApp es privada, y no podíamos hacer uso de ninguna aplicación gratuita de envío de mensajes. Sin embargo, este problema ya tiene solución, mediante el uso de la API del nuevo Google Hangouts, que llega tarde como para incluirlo en la aplicación.

Durante el desarrollo de la aplicación hubo un cambio en la versión de la API de Google Maps, pasando de la versión 1 a la versión 2. Tuvimos que rehacer la parte correspondiente a estos cambios, ya que las mejoras en la versión 2 nos proporcionaban mayor facilidad en la programación y la primera versión pasaba a estar obsoleta, dejando de proporcionar claves de acceso.

Una de las principales preocupaciones de los padres de los niños era la seguridad de la aplicación, y que los datos proporcionados no pudieran proporcionar ninguna información a alguien con malos propósitos, con lo cual tuvimos que adaptarnos a esta situación y solicitar sólo los datos necesarios. Esto generaba un pequeño problema a la hora de generar el camino que cada niño sigue desde su casa hasta el colegio, ya que al no tener la dirección de la vivienda no podíamos fijar el punto de inicio de la ruta ahí. La solución de compromiso fue empezar la ruta desde la ubicación actual del niño, estableciendo el requisito de estar en su casa o en el colegio a la hora de enviar su camino seguro. De esta forma el niño deberá dibujar sobre el mapa el camino desde su casa al colegio o desde el colegio a su casa, dependiendo de donde se encuentre.



Durante todo el desarrollo de la aplicación *Camino seguro al cole* se ha llevado un control por parte del Ayuntamiento de Madrid, como puede verse en el registro de reuniones de este mismo documento. Ha sido la primera vez que el cliente es alguien externo a la facultad, y hemos trabajado en un marco que nos ha permitido hacernos una idea de lo que es el mundo laboral, aunque siempre hemos contado con la ayuda de los directores del proyecto, a los cuales debemos agradecer la ayuda proporcionada durante todo el año.

En el siguiente apartado se especificarán las futuras líneas de trabajo, que pueden extender la funcionalidad de la aplicación y, que por falta de tiempo o requisitos no se han podido incluir en el desarrollo de la aplicación.

## **5.2. Futuras líneas de trabajo**

### **Adaptación del servidor y la base de datos a los estándares del IAM**

Para la distribución de la aplicación *Camino seguro al cole* es necesaria la adaptación del servidor y la base de datos a los estándares del IAM<sup>[8]</sup>.

Durante el desarrollo de la aplicación, para realizar las pruebas, se desplegaron los nodos del servidor y la base de datos sobre un servidor gratuito (000webhost) sobre el cual se podía utilizar el lenguaje *PHP* para las transferencias entre el servidor y la base de datos.

Para la adaptación hay que seguir los pasos descritos en la guía de estándares del IAM, en la sección diseño de arquitectura lógica. La guía se puede consultar en la referencia correspondiente.

### **Automatización del proceso de visualización de todos los caminos seguros de un colegio**

Para la visualización de todos los caminos seguros de un colegio debemos seguir la sección *Obtención de la araña de movilidad de un colegio* de los anexos de este documento. Este proceso se podría automatizar haciendo uso de la API de Google Maps versión 3 de Javascript.

Para ello habría que almacenar en una estructura el registro de todos los caminos seguros de un colegio, que se puede obtener siguiendo los pasos del apartado *Obtener un registro con todas las rutas pertenecientes a un colegio* del anexo 5 de este documento. Una vez almacenado en una estructura habría que recorrer la lista de puntos *LatLng* que proporcionan la ruta de cada usuario, y dibujar el trayecto sobre el mapa utilizando un color diferente para la ruta de cada niño (haciendo uso de la clase *Polyline* de la API de Google Maps). La documentación de esta API<sup>[9]</sup> se puede consultar en la referencia correspondiente.

Una vez dibujados todos los caminos seguros de un colegio sobre el mapa se procederá al establecimiento de la araña de movilidad para dicho colegio. El proceso de generación de la araña de movilidad también es manual, a pesar de la utilización de la herramienta Google Earth. Sería muy interesante la automatización de este proceso, ya que ahorraría mucho tiempo en la generación de la araña de movilidad de un colegio. Para ello se podría usar el modelo matemático que se expone en el siguiente punto: *Optimización de rutas en el camino seguro al cole*.

## **Optimización de rutas en el camino seguro al cole**

El criterio para el establecimiento de la araña de movilidad en el momento del desarrollo de la aplicación era seguir los caminos por los que fueran el mayor número de niños, pero este criterio podría variar y establecer una araña de movilidad distinta según distintos criterios. De forma concurrente con el desarrollo de la aplicación móvil se desarrolló un modelo matemático llevado a cabo por dos alumnas de la facultad de matemáticas. Este modelo matemático tiene como objetivo la optimización de rutas en el *Camino seguro al cole*. Hay dos vías de estudio distintas.

Rocío Colina generó un modelo matemático para proponer rutas alternativas según una valoración de la seguridad en el *Camino seguro al cole*. Esta valoración de la seguridad se hace con diferentes criterios. Beatriz Municio hizo un modelo de comparación de rutas con los transportes alternativos (autobús, coche o bicicleta). Ambos estudios forman parte de los trabajos de fin de grado de las respectivas alumnas, y los proyectos han sido dirigidos por la profesora Inés María Gómez Chacón.

La incorporación de estos modelos matemáticos al panel de administración junto con la automatización de la visualización de todos los caminos seguros de un colegio facilitarían considerablemente la generación de las arañas de movilidad. Para obtener más información se pueden consultar los trabajos de fin de grado de estas alumnas en la facultad de matemáticas.

## **Uso de la API de Google Hangouts para el envío de notificaciones**

A finales de mayo de 2013, Google renovó el sistema antes conocido como Google Talks presentando Google Hangouts, un sistema de mensajería instantánea. La API de este sistema de mensajería es abierta, con lo que podríamos hacer uso de ella para el envío de mensajes y notificaciones. La documentación de esta API<sup>[10]</sup> se puede consultar en la referencia correspondiente.



# Bibliografía

Roger S. Pressman & Associates, Inc.: Software Engineering: A Practitioner's Approach (7th Edition).

ISBN-10: 0071267824. ISBN-13: 978-0071267823. McGraw-Hill, 2009.

Guía para desarrolladores, Android Developers

< <http://developer.android.com/guide/components/index.html> >

Blog sobre programación < <http://www.sgoliver.net/blog/> >

# Referencias

[1] "Gasolineras España" <<http://www.mobialia.com/apps/gas-stations/>>

[2] "Farmacias Ahora! Zaragoza"

<<https://play.google.com/store/apps/details?id=com.geoslab.farmaciasahorazgz>>

[3] "SAFA"

<[https://play.google.com/store/apps/details?id=appinventor.ai\\_manugalan102.FundacionSAFA&feature=search\\_result#?t=W251bGwsMSwxLDEsImFwcGludmVudG9yLmFpX21hbnVnYWxhbjEwMi5GdW5kYWNPb25TQUZBIi0](https://play.google.com/store/apps/details?id=appinventor.ai_manugalan102.FundacionSAFA&feature=search_result#?t=W251bGwsMSwxLDEsImFwcGludmVudG9yLmFpX21hbnVnYWxhbjEwMi5GdW5kYWNPb25TQUZBIi0)>

[4] "Amber GPS Alert" <<http://www.amberalertgps.com/>>

[5] "Waze" <<http://es.waze.com/>>

[6] "Números de alumno por aula en enseñanzas no universitarias reguladas en la LOE" <<http://www.educa2.madrid.org/documents/397121/2959924/08.+N%C3%BAmero+m%C3%A1ximo+de+alumnos+por+aula.pdf/e62f1288-169e-463b-a1b8-869f26e19462> >

[7] "Fórmula de Haversine" < [http://en.wikipedia.org/wiki/Haversine\\_formula](http://en.wikipedia.org/wiki/Haversine_formula) >

< <http://www.longitudestore.com/haversine-formula.html> >

[8] "Guía de estándares del IAM"

<<http://www.madrid.es/UnidadesDescentralizadas/UDCIAM/Documentaci%C3%B3n%20t%C3%A9cnica/Gu%C3%ADaEst%C3%A1ndares/VersionMadrid/Gu%C3%ADa%20de%20Est%C3%A1ndares.pdf> >

[9] "API de Google Maps para Javascript"

<<https://developers.google.com/maps/documentation/javascript/?hl=es>>

[10] "API de Google Hangouts"

<<https://developers.google.com/+hangouts/api/?hl=es>>

[11] "Distribución actual de las versiones de Android"

<<http://developer.android.com/about/dashboards/index.html>>

[12] "Conversión de unidades angulares con Excel"

< <http://support.microsoft.com/kb/213449/es> >



# Anexos

## Anexo 1. Definiciones, acrónimos y abreviaturas

Araña de movilidad: es la representación gráfica del *Camino seguro al cole*, está formado por el entramado de calles más frecuentadas por los escolares.

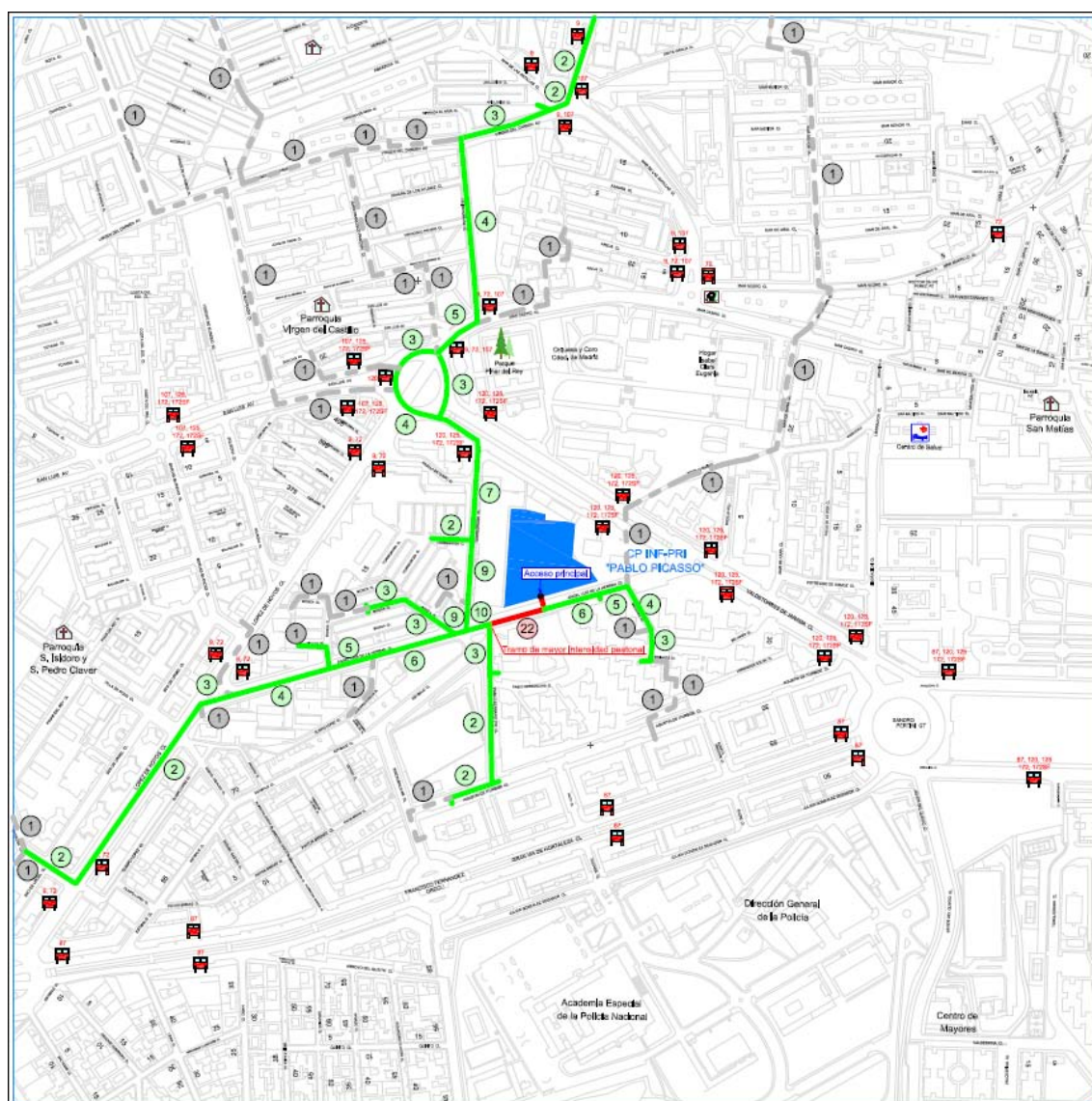


Figura 12: Araña de movilidad para el colegio CEIP "Pablo Picasso".

Comercio amigo de los niños: es un comercio adscrito al proyecto *Camino seguro al cole*, tienen una pegatina identificativa en la puerta del comercio. Su función es vigilar y proteger a los niños de cualquier incidente en el *Camino seguro al cole*.



Figura 13: Pegatina identificativa de comercio amigo de los niños.

## Anexo 2. Distribución actual de las versiones de Android

El siguiente gráfico representa el uso de las diferentes versiones de Android<sup>[11]</sup>, y está basado en el acceso de los dispositivos a Google Play durante un periodo de 14 días finalizado el 4 de Febrero de 2013.

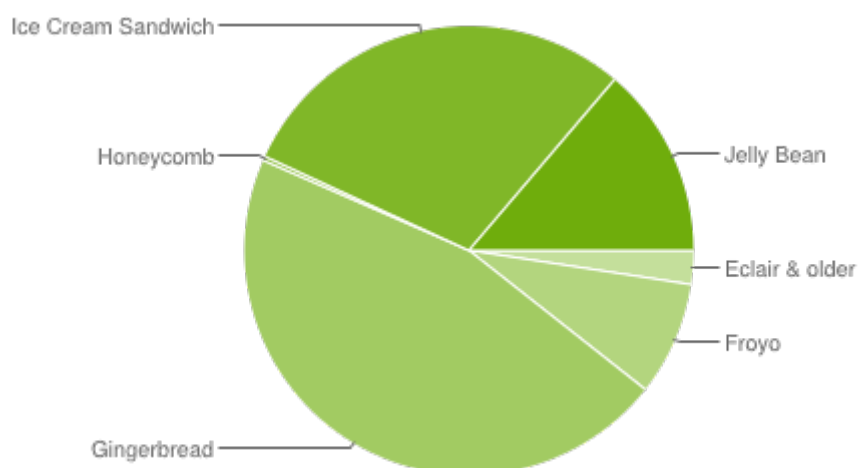


Figura 14: Distribución actual de las versiones de Android.

La siguiente tabla indica los porcentajes de uso de cada distribución.

Versión	Nombre	API	Porcentaje
1.6	Donut	4	0.2%
2.1	Eclair	7	2.2%
2.2	Froyo	8	8.1%
2.3 - 2.3.2	Gingerbread	9	0.2%
2.3.3 - 2.3.7		10	45.4%
3.1	Honeycomb	12	0.3%
3.2		13	1.0%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	29.0%
4.1	Jelly Bean	16	12.2%
4.2		17	1.4%



### **Anexo 3. Contenido de las páginas PHP referenciadas desde la clase Llamada**

#### **Conexion.php**

```
<?PHP
// Fichero con los datos de conexion a la BBDD
function Conectarse()
{
    //net
    $db_host="*****"; //no se muestran por seguridad
    $db_nombre="*****";
    $db_user="*****";
    $db_pass="*****";

    $link=mysql_connect($db_host, $db_user, $db_pass);
    if(link){
        mysql_select_db($db_nombre, $link);
    }
    return $link;
}
?>
```

#### **Token.php**

```
<?PHP
function getToken($length = "6")
{
    $token = md5(uniqid(rand(), true));
    if ($length != "") return substr($token, 0, $length);
    else return $token;
}
?>
```

#### **Login.php**

```
<?php

//Comprueba que la contraseña coincide con el usuario

include("conexion.php");
include("token.php");

$user = $_POST['user'];
$pass = $_POST['pass'];
```

```
// Conectar con el servidor de base de datos
$con = Conectarse();

$user_safe= mysql_real_escape_string($user);
$pass_safe = mysql_real_escape_string($pass);
$instruccion="SELECT pass, numUs FROM us WHERE numUs='$user_safe'";

if ($result = mysql_query($instruccion,$con)){

    if($row = mysql_fetch_array($result)){
        if($row["pass"] == $pass_safe){
            echo "1"; //todo OK
        }else{
            echo "2"; //pass erronea
        }
    }else{
        echo "3"; //user no existe
    }
}else{
    echo "4"; //error conexion bd
}
mysql_close($con);
?>
```

### **Register.php**

```
<?php

//Registro de un nuevo usuario
include("conexion.php");

$user = $_POST['user'];
$pass = $_POST['pass'];
$nombre = $_POST['nombre'];
$direccion = $_POST['dir'];
$tutor = $_POST['tutor'];
$tlfTutor = $_POST['tlfTutor'];
$colegio= $_POST['colegio'];

$con = Conectarse();

$user_safe= mysql_real_escape_string($user);
$pass_safe = mysql_real_escape_string($pass);
$nombre_safe = mysql_real_escape_string($nombre);
$dir_safe = mysql_real_escape_string($direccion);
$tutor_safe = mysql_real_escape_string($tutor);
$tlfTutor_safe = mysql_real_escape_string($tlfTutor);
```

```

$colegio_safe = mysql_real_escape_string($colegio);

if ($Consulta = mysql_query("INSERT INTO
us(numUs,pass,nombre,dir,tutor,tlfTutor,nombCol)
VALUES('$user_safe','$pass_safe','$nombre_safe','$dir_safe','$tutor_safe','$tlfT
utor_safe','$colegio_safe')",$con))
{
    echo "1";
}else{
    echo "2";
}
mysql_close($con);
?>

```

### **GetColegio.php**

```

<?php

//Llamada que devuelve todos los datos relacionads con un colegio dado

include("conexion.php");

$colegio = $_POST['colegio'];

// Conectar con el servidor de base de datos
$con = Conectarse();

$colegio_safe= mysql_real_escape_string($colegio);

$instruccion="SELECT dir ,tlf,coord, numEst FROM Colegios WHERE
nombCol='$colegio_safe'";

if ($result = mysql_query($instruccion,$con)){

    if($row=mysql_fetch_assoc($result)){
        $output[]=$row;
        while($row=mysql_fetch_assoc($result))
            $output[]=$row;
        print(json_encode($output));
    }else{
        echo "3"; //El colegio no existe
    }
}else{
    echo "4"; //error conexion bd
}
mysql_close($con);
?>

```

### **GetArania.php**

```
<?php
//Llamada que devuelve las coordenadas comienzo y destino de todas y cada
una de las calles que forman la araña de movilidad del colegio introducido.
include("conexion.php");

$colegio = $_POST['colegio'];

// Conectar con el servidor de base de datos
$con = Conectarse();

$colegio_safe= mysql_real_escape_string($colegio);

$instruccion="SELECT origen , destino FROM Arañas WHERE
nombCol='$colegio_safe'";

if ($result = mysql_query($instruccion,$con)){

    if($row=mysql_fetch_assoc($result)){
        $output[]=$row;
        while($row=mysql_fetch_assoc($result))
            $output[]=$row;
        print(json_encode($output));
    }else{
        echo "3"; //El colegio no existe
    }
}else{
    echo "4"; //error conexion bd
}
mysql_close($con);
?>
```

### **GetEstablecimiento.php**

```
<?php

include("conexion.php");

//Llamada que devuelve todos los datos de los establecimientos
correspondientes a un colegio dado.

$colegio = $_POST['colegio'];

// Conectar con el servidor de base de datos
$con = Conectarse();
```

```

$colegio_safe= mysql_real_escape_string($colegio);

$instruccion= "SELECT E.nombEst, E.dir, E.coord,E.nombCol, E.valoracion FROM
Establecimientos E WHERE E.nombEst IN (SELECT A.nombEst FROM Adheridos
A where A.nombCol = '$colegio_safe') and E.nombCol = '$colegio_safe'";

if( $result = mysql_query($instruccion,$con)){

    if($row=mysql_fetch_assoc($result)){
        $output[] = array_map(htmlentities,$row);

        while($row=mysql_fetch_assoc($result)){
            $output[] = array_map(htmlentities,$row);
        }

        echo json_encode($output);

    }else{
        echo "3"; //El colegio no existe
    }
}else{
    echo "4"; //Error al conectar con la base de datos
}
mysql_close($con);
?>

```

### **SetArania.php**

```

<?php

//Permite insertar las coordenadas de una nueva calle en el colegio
seleccionado

include("conexion.php");

$colegio = $_POST['colegio'];
$origen = $_POST['origen'];
$destino= $_POST['destino'];

$con = Conectarse();

$colegio_safe= mysql_real_escape_string($colegio);
$origen_safe = mysql_real_escape_string($origen);
$destino_safe = mysql_real_escape_string($destino);

if ($Consulta = mysql_query("INSERT INTO Arañas(nombCol,origen,destino)
VALUES('$colegio_safe','$origen_safe','$destino_safe')",$con))

```

```

{
    echo "1";
}else{
    echo "2";
}
mysql_close($con);
?>

```

### **SetRuta.php**

```
<?php
```

//Permite insertar una nueva ruta para un determinado colegio y alumno

```
include("conexion.php");
```

```

$user = $_POST['user'];
$ruta = $_POST['ruta'];
$colegio= $_POST['colegio'];

```

```
$con = Conectarse();
```

```

$user_safe= mysql_real_escape_string($user);
$ruta_safe = mysql_real_escape_string($ruta);
$colegio_safe = mysql_real_escape_string($colegio);

```

```

if ($Consulta = mysql_query("INSERT INTO Ruta(numUs,ruta,nombCol)
VALUES('$user_safe','$ruta_safe','$colegio_safe')",$con))
{
    echo "1";
}else{
    echo "2";
}
mysql_close($con);
?>

```

### **SetValoracion.php**

```
<?php
```

//Llamada que permite a los usuarios valorar un establecimiento

```
include("conexion.php");
```

```

$establecimiento= $_POST['establecimiento'];
$user = $_POST['user'];
$dir = $_POST['dir'];

```

```

$puntuacion= $_POST['puntuacion'];

$con = Conectarse();

$establecimiento_safe = mysql_real_escape_string($establecimiento);
$user_safe= mysql_real_escape_string($user);
$dir_safe = mysql_real_escape_string($dir);
$puntuacion_safe = mysql_real_escape_string($puntuacion);

if ($Consulta1 = mysql_query("INSERT INTO
valoraciones(nombEst,dir,numUs,puntuacion)
VALUES('$establecimiento_safe','$dir_safe','$user_safe','$puntuacion_safe')",$
con))
{
    if ($result = mysql_query("SELECT AVG(puntuacion) from valoraciones where
nombEst='$establecimiento_safe' and dir ='$dir_safe'", $con))
    {
        if($row=mysql_fetch_assoc($result)){
            $media[]=$row;
            $media2 = json_encode($media);
        }

        if($Consulta2 = mysql_query("UPDATE Establecimientos SET
valoracion='$media2[21]' WHERE nombEst ='$establecimiento_safe' and
dir='$dir_safe'", $con))
        {
            echo "1";

        }else{
            echo "2";
        }
    }else{
        echo "2";
    }
}
mysql_close($con);
?>

```

## Anexo 4. Manual de usuario

### Introducción

En este manual se detallará la instalación y el uso de la aplicación *Camino seguro al cole*, con la ayuda de imágenes para facilitar la comprensión. Al ser una aplicación cuyos usuarios van a ser niños (con la ayuda de sus padres) se ha intentado que el manejo sea simple e intuitivo, haciendo uso de los botones del dispositivo para mostrar opciones o para volver al menú principal. La aplicación *Camino seguro al cole* está desarrollada para el sistema operativo Android en la versión 4.0 “Ice Cream Sandwich”, la instalación en sistemas operativos no compatibles puede producir errores.

Las imágenes mostradas son capturas de pantalla realizadas en un Sony Ericson Xperia Neo V (MT11i) con el sistema operativo Android 4.0.4, la apariencia y ubicación de los menús puede variar ligeramente con respecto a su dispositivo.

### Instalación

1. Descargar el archivo *Camino.apk* de la aplicación *Camino seguro al cole* de la página web [www.tecnologiaucm.es](http://www.tecnologiaucm.es). Podemos descargar el archivo desde el terminal móvil o desde un ordenador.
2. En el móvil debemos permitir la instalación de aplicaciones que no son de Market. Para ello hay que entrar en el menú ajustes del móvil, acceder al submenú *Seguridad* y activar la casilla *Orígenes desconocidos*. Se nos informará de los riesgos que conlleva instalar una aplicación de origen desconocido y tendremos que aceptar la política de responsabilidad. No instale aplicaciones cuyo origen no sea fiable.

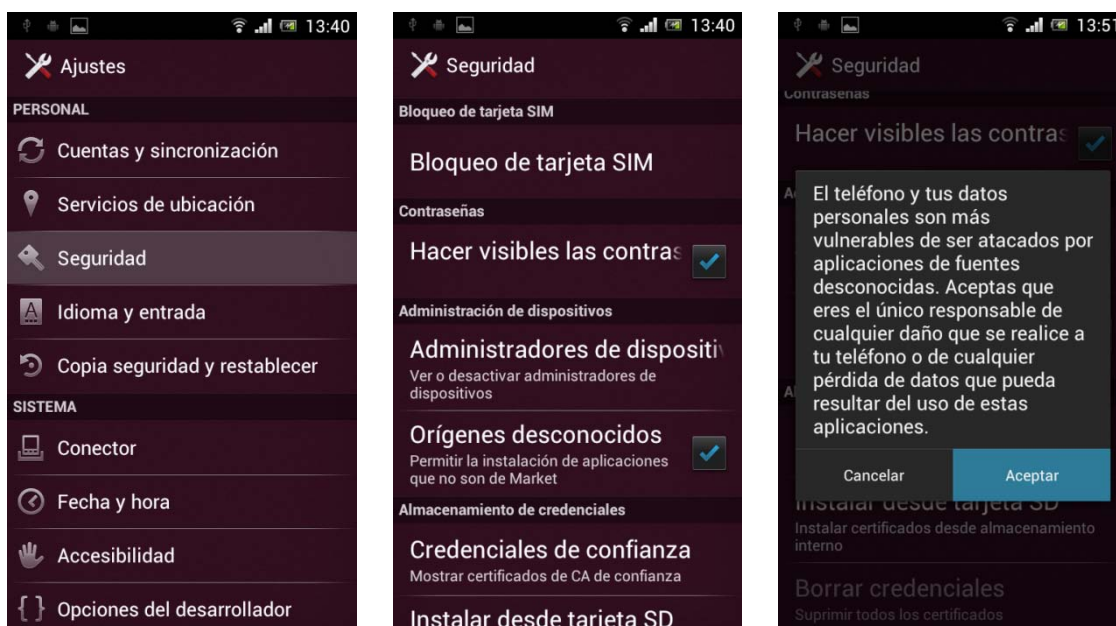




Figura 15: Configuración de seguridad para la instalación de la aplicación.

3. Colocar el archivo *Camino.apk* en la tarjeta SD del terminal móvil.
4. En el móvil, abrir un explorador de archivos, buscar el archivo *Camino.apk* en la tarjeta SD del y pulsar sobre él. Se nos informará de los permisos que tendrá la aplicación y pulsando el botón *Aceptar* comenzará la instalación. Una vez instalada podemos elegir entre abrirla inmediatamente o salir.

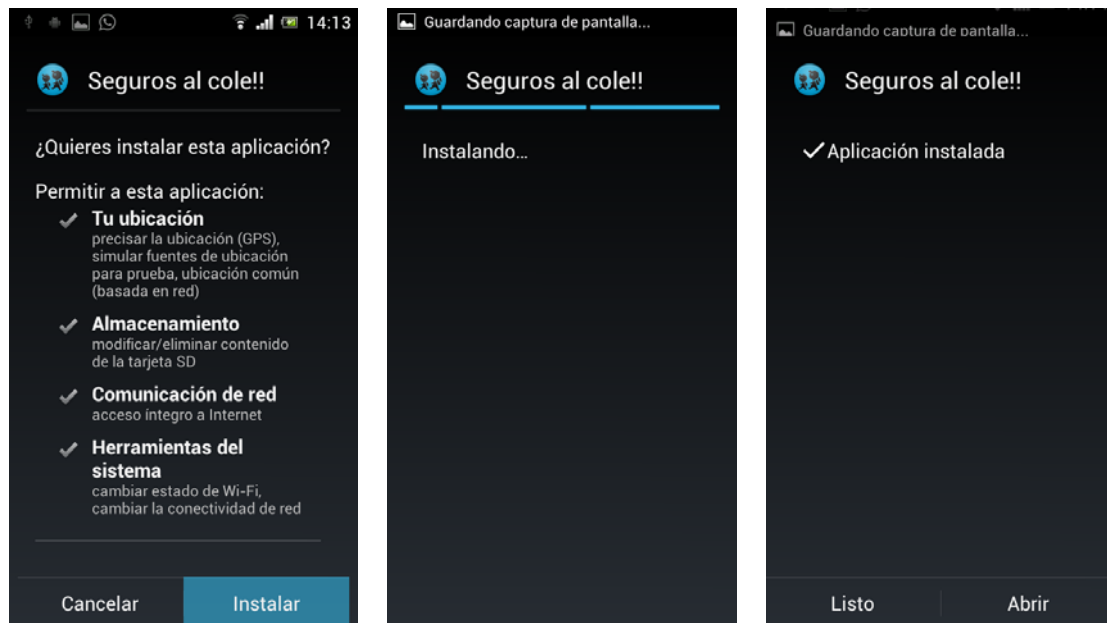


Figura 16: Instalación de la aplicación *Camino seguro al cole*.

5. Para iniciar la aplicación pulsar sobre el icono del menú principal Seguros al cole!!.

## Inicio

Una vez instalada la aplicación, se puede iniciar en cualquier momento pulsando sobre el icono del menú principal. Al iniciar la aplicación se nos mostrará una breve pantalla de introducción, y después se mostrará la pantalla inicial de la aplicación *Camino seguro al cole*, desde la que se accede al menú principal.

Si no estamos registrados en la aplicación, este será el primer paso que debemos hacer.

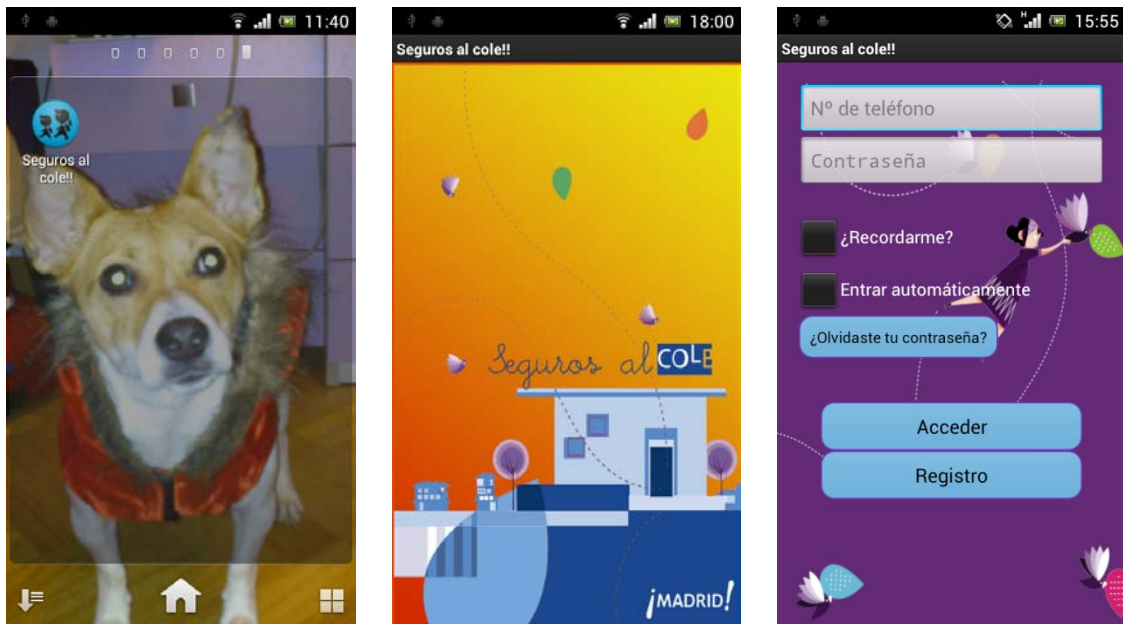


Figura 17: Inicio de la aplicación *Camino seguro al cole*.

## Registro

Para acceder a las funcionalidades de la aplicación es necesario estar registrados en la base de datos de la aplicación *Camino seguro al cole*.

Para registrar un nuevo usuario hay que pulsar el botón registro de la pantalla de inicio de la aplicación y rellenar correctamente todos los campos del formulario. También debemos aceptar la política de privacidad en la que se nos informa de que nuestros datos van a ser introducidos en una base de datos.

Los campos del formulario son los siguientes:

- Nº de teléfono: el número de teléfono móvil en el que está instalada la aplicación.
- Nombre: sólo se usa en el menú principal para mostrar un saludo, puede ser un nombre o un pseudónimo.
- Contraseña: contraseña personal que dará acceso al menú principal de la aplicación. Distingue entre letras mayúsculas y minúsculas.
- Repetir contraseña: escribir la contraseña de nuevo, de forma que coincida con el campo anterior.
- Colegio: habrá que seleccionar de la lista desplegable el colegio en el que el niño esté matriculado.
- Código colegio: introducir el código proporcionado por el colegio para el registro en la aplicación *Camino seguro al cole*. Debe coincidir con el código almacenado en la base de datos para el colegio seleccionado.

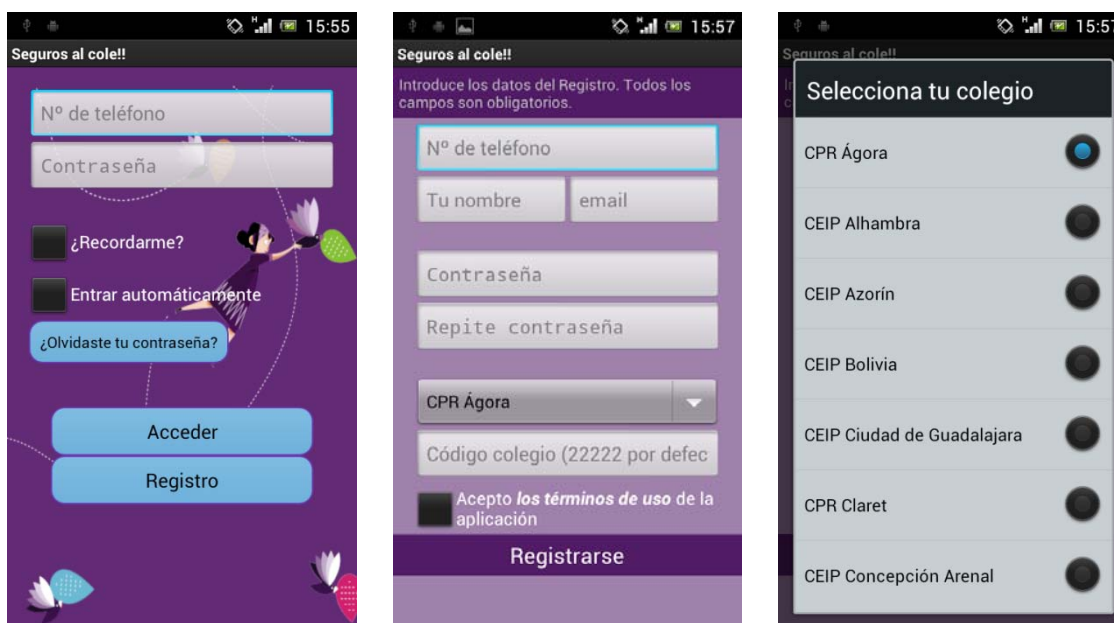


Figura 18: Registro en la aplicación *Camino seguro al cole*.

## Identificación

Una vez realizado el proceso de registro, cada vez que queramos acceder a la aplicación tendremos que identificarnos en la pantalla de inicio de la aplicación. Para ello debemos rellenar los campos N° de teléfono y Contraseña de forma que coincidan con la información proporcionada durante el registro. Pulsando el botón *Acceder* entramos en el menú principal de la aplicación.

Opcionalmente podemos marcar las casilla *¿Recordarme?* para no tener que introducir los datos cada vez que abrimos la aplicación. Si marcamos la casilla *Entrar automáticamente* el proceso de identificación se hará de forma invisible al usuario. En caso de que nos olvidemos de la contraseña, el botón *¿Olvidaste tu contraseña?* enviará un email a la dirección proporcionada durante el registro con una nueva contraseña.

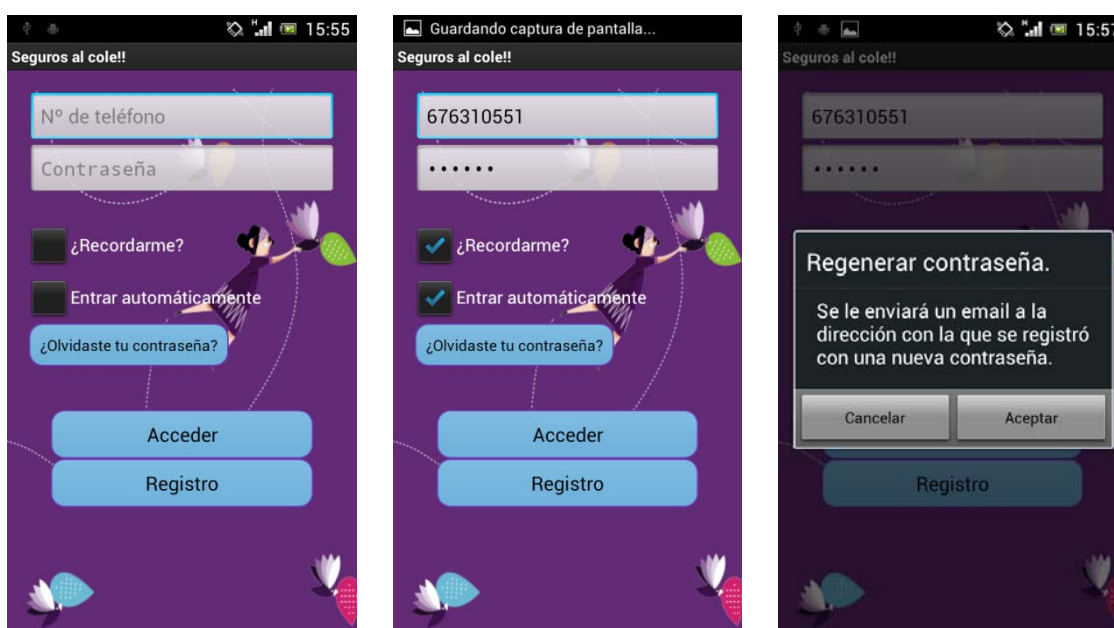


Figura 19: Identificación en la aplicación Camino seguro al cole.

## Menú principal

Desde el menú principal podemos acceder a todas las funcionalidades de la aplicación pulsando sobre los distintos botones. Dentro de cada una de las funcionalidades pulsando el botón atrás del dispositivo se vuelve al menú principal.



Figura 20: Menú principal de la aplicación.

## Mi ruta segura

Desde el primer botón de la aplicación podemos consultar la araña de movilidad (conjunto de caminos seguros) para el colegio seleccionado durante el registro, así como localizarnos en un mapa y guiarnos desde el punto en que nos encontremos hasta el punto más cercano de entrada a la araña de movilidad.

Pulsando el botón de opciones del dispositivo se despliega un menú con las siguientes opciones:

- *Mostrar cómo llegar a la ruta segura:* trazará en color azul la ruta a pie más rápida desde donde esté el niño hasta el punto más cercano de la araña de movilidad.
- *Navegar libremente por el mapa:* permite mover la vista sobre el mapa sin restricciones. Esta opción está activada por defecto.
- *Fija el mapa sobre tu posición actual:* centra la vista del mapa sobre la posición actual del niño.



Sobre el mapa podemos utilizar todos los controles que permite Google Maps, como acercar o alejar el zoom, o girar el mapa. Pulsando sobre los marcadores podemos acceder al nombre del colegio y a los nombres de los establecimientos amigos de los niños, además de visualizar su posición en el mapa.

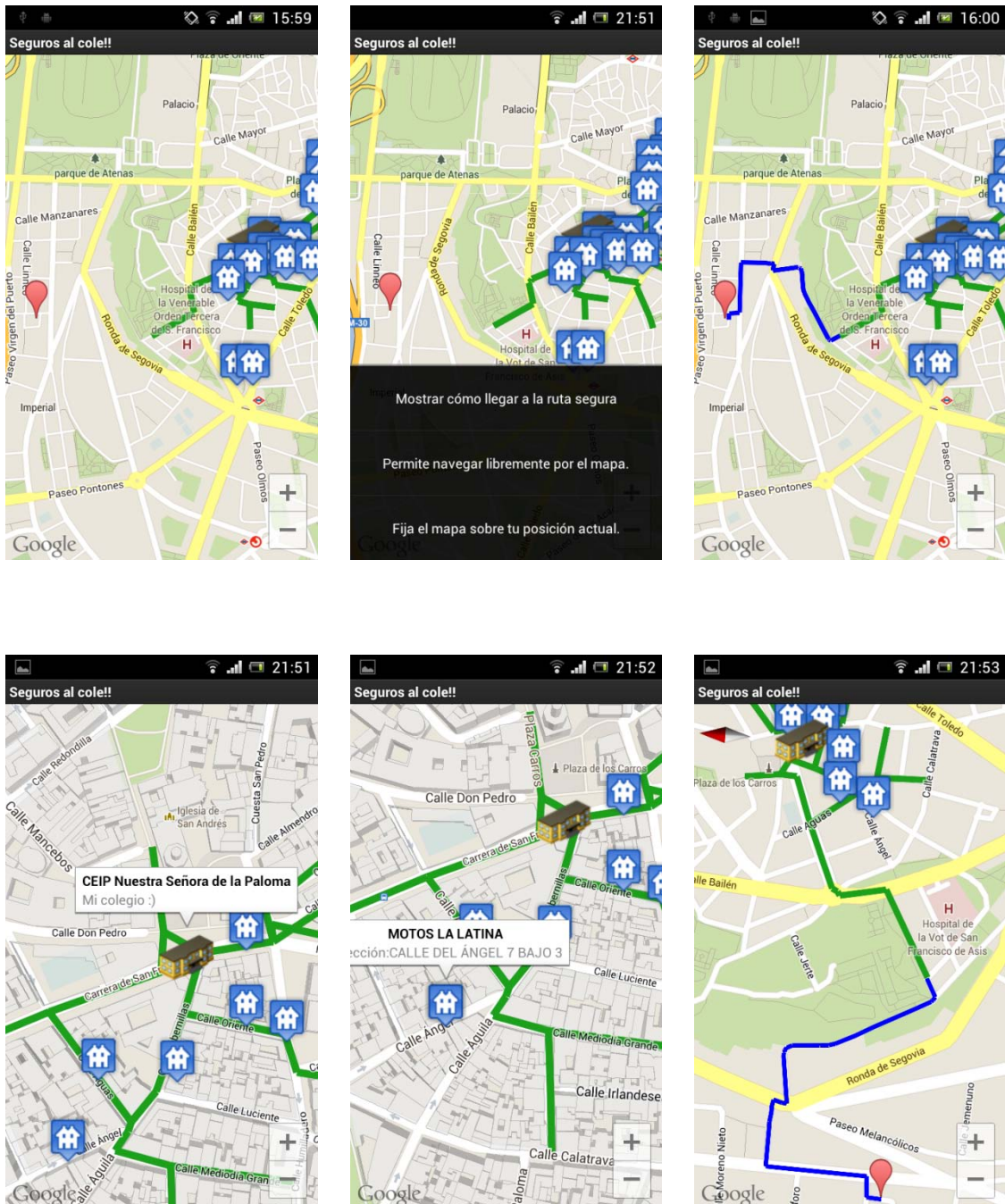


Figura 21: Mi ruta segura.

## Crear mi ruta al cole

El botón *Crear mi ruta al cole* permite la recogida de la ruta que sigue un alumno desde su casa hasta el colegio, para la creación o actualización de la araña de movilidad.

El marcador rojo indica nuestra posición y el marcador con el dibujo del edificio indica el colegio. El punto de partida es la posición actual del dispositivo, por lo que, para subir nuestra ruta al cole debemos hacerlo desde el colegio o desde nuestra casa dibujando el trayecto colegio-casa o casa-colegio respectivamente.

Para dibujar el trayecto hay que ir trazando líneas pulsando punto a punto en el mapa sobre las calles por las que vamos al colegio.

Pulsando el botón de opciones del dispositivo se despliega un menú con las siguientes opciones:

- *Deshacer el último tramo*: borra la última línea añadida a la ruta.
- *Volver a empezar*: borra todas las líneas creadas y empieza la ruta desde el principio (la posición actual del dispositivo).
- *Enviar*: sube la ruta a la base de datos. Pulsar sólo cuando la ruta dibujada sea la correcta.

La precisión en el dibujo de la ruta debe ser lo suficientemente buena como para que no haya confusiones sobre las calles involucradas.

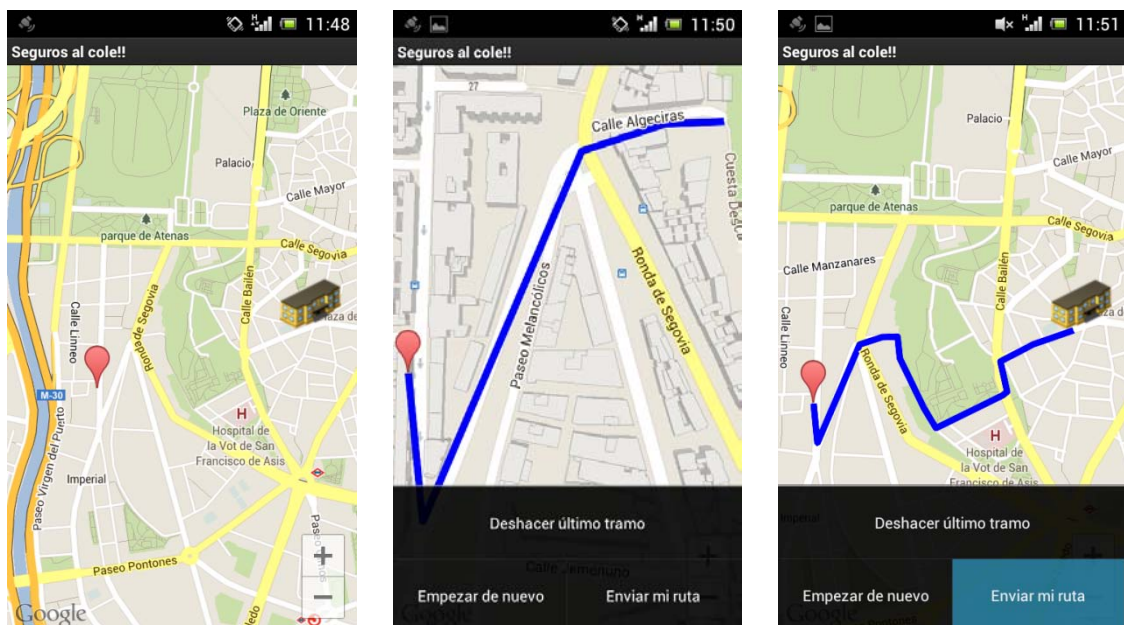


Figura 22: Crear mi ruta al cole.

## Reportar incidencia

El botón *Reportar incidencia* permite notificar cualquier incidencia que halla en el camino seguro directamente al Ayuntamiento de Madrid. Para ello nos redirige a la página web de gestiones y trámites del Ayuntamiento de Madrid. Desde esta página podemos tramitar en línea, por teléfono o por correo electrónico cualquier incidencia buscando el aviso correspondiente de la lista.

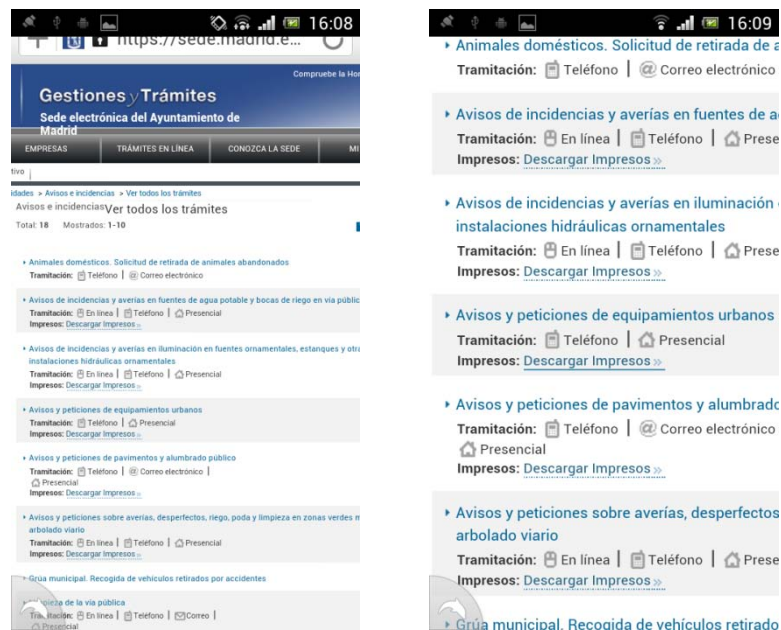


Figura 23: Reportar incidencia.



## Valorar comercios

El botón *Valorar comercios* permite hacer una valoración de cero a cinco estrellas (en intervalos de media estrella) de un comercio amigo de los niños.

Al pulsar el botón *Valorar comercios* del menú principal se nos muestra una lista de todos los comercios amigos de los niños asignado al colegio seleccionado en el registro, junto con su valoración media.

Los comercios que ya han sido valorados por nosotros aparecen en color gris, y en color negro los que aún no hemos valorado. Sólo se permite una valoración por persona y comercio para evitar falsear las valoraciones medias, aunque sí es posible modificar nuestra valoración anterior.

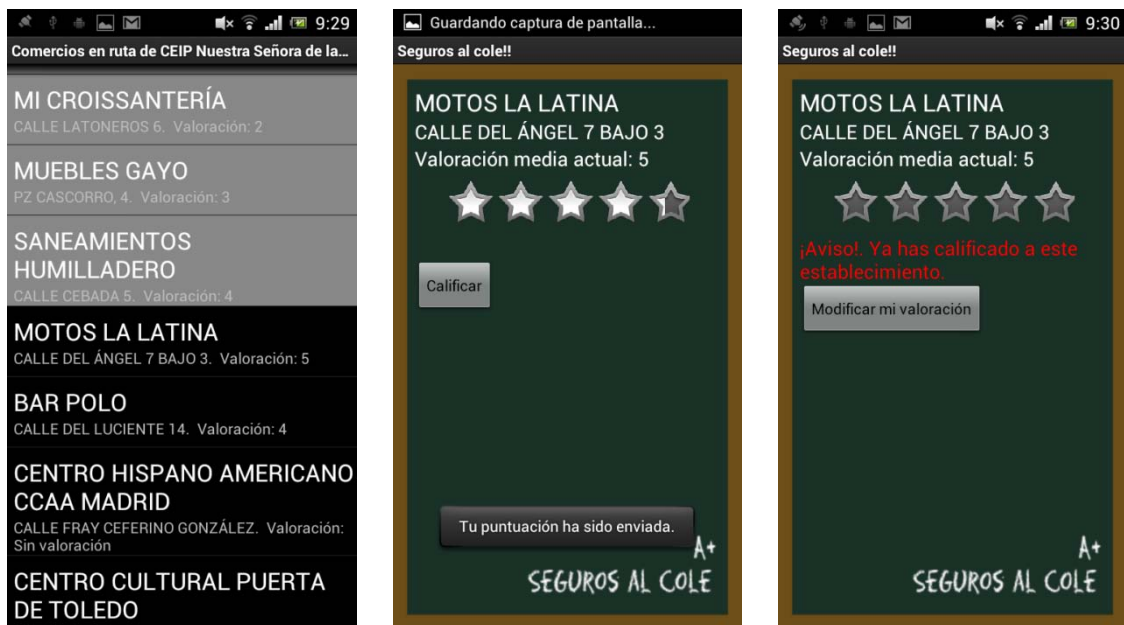


Figura 24: Valorar comercios.

## Opciones

El botón *Opciones* permite seleccionar un colegio, cambiar de usuario y mostrar un texto de ayuda explicando el manejo de la aplicación.

Las dos primeras opciones están pensadas para la actualización de la araña de movilidad. La idea es llevar la aplicación *Camino seguro al cole* instalada en una tableta a los colegios, y que los niños de las clases se lo pasen cambiando de usuario y subiendo sus rutas seguras.

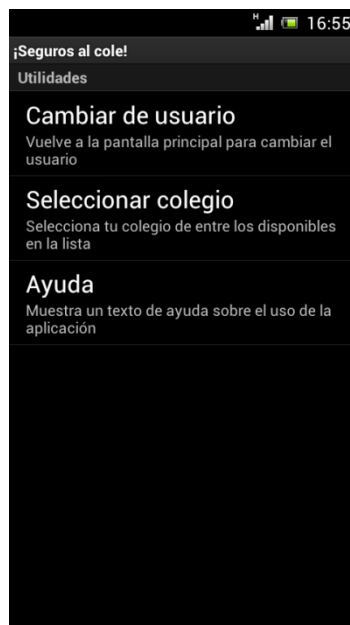


Figura 25: Opciones.

## Anexo 5. Manual de administración

### Introducción

En este manual se detallará el manejo del panel de administración, desde el cual se pueden gestionar los datos contenidos en la base de datos para la aplicación *Camino seguro al cole*.

Para acceder al panel el administrador debe estar dado de alta en el sistema como administrador, para más información consultar el caso de uso *Añadir nuevo administrador*.

En el momento de la redacción de este documento tanto la base de datos como el panel de administración estaban alojados en un servidor gratuito, que podría o no ser el actual servidor, por tanto las direcciones especificadas en este documento podrían variar.

### Inicio

El acceso se realiza a través de la siguiente dirección url: <http://www.segurosalcole.comoj.com/administracion/login.php>

En esta página de identificación debemos introducir nuestros datos de forma que coincidan con los datos que dimos en el registro de nuevo administrador y pulsar el botón *Entrar*.

### Introduzca sus datos para acceder al panel de administración.

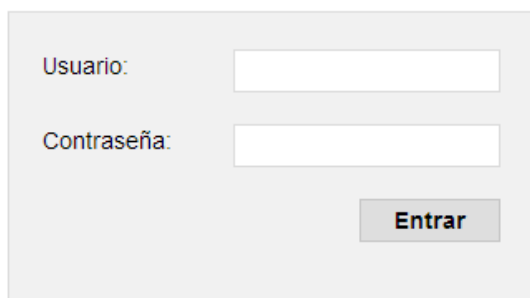
El formulario de inicio de sesión está contenido en un recuadro gris claro. Incluye dos campos de entrada de texto: el superior está etiquetado como 'Usuario:' y el inferior como 'Contraseña:'. A la derecha de cada etiqueta hay un campo de texto blanco. Debajo de estos campos, a la derecha, hay un botón rectangular gris con el texto 'Entrar' en blanco.

Figura 26: Pantalla de inicio del panel de administración.

Una vez introducidos los datos correctamente accederemos a la siguiente página, desde la cual podemos insertar un nuevo colegio, insertar la araña de movilidad, insertar un establecimiento, añadir un administrador, obtener las rutas subidas por los usuarios de un colegio y obtener registros con todos los colegios y establecimientos de la base de datos de la aplicación *Camino seguro al cole*.

\*\*\*\*\* **PANEL DE ADMINISTRACION** \*\*\*\*\*

### Insertar nuevo colegio

Nombre:  Ej: CEIP Rosalía de Castro  
Coordenadas:  Ej: 40.474375,-3.867187  
Dirección:  Ej: C/Rosalía de Castro 20  
Codigo:  Ej: 22222  
Telefono:  Ej: 123456789

### Insertar calle en araña de movilidad

Nombre del colegio:  Ej: CEIP Rosalía de Castro  
Coordenada de origen:  Ej: 40.474375,-3.867187  
Coordenada destino:  Ej: 40.474375,-3.867187

### Insertar establecimiento

Nombre del establecimiento:  Ej: Farmacia  
Dirección:  Ej: C/Comunidad de Brasil  
Coordenadas:  Ej: 40.474375,-3.867187  
Nombre del colegio:  Ej: CEIP Rosalía de Castro  
Valoración:  La valoración puede ser desde 0 hasta 5 en intervalos de 0,5

### Obtener un registro con todas las rutas pertenecientes a un colegio.

Nombre del colegio:  Ej: CEIP Rosalía de Castro

### Obtener un registro con todos los colegios almacenados.

### Obtener un registro con todos los establecimientos adheridos.

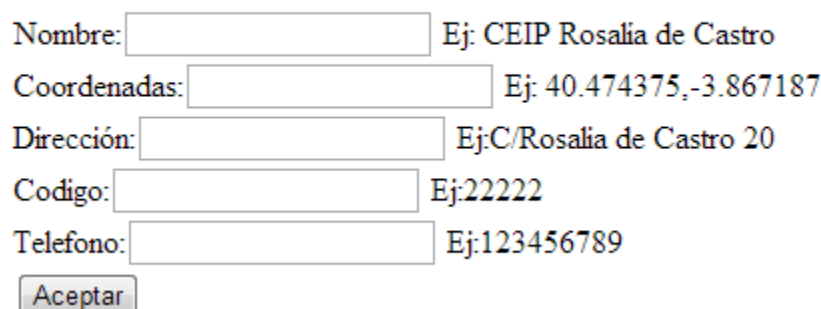
Para cerrar la sesión puedes hacer un [logout](#)

Figura 27: Panel de administración.

## Insertar un nuevo colegio

Podemos insertar un nuevo colegio en la tabla de colegios de la base de datos de la aplicación *Camino seguro al cole*. Para ellos tenemos que rellenar los campos del formulario proporcionando la información nombre, dirección, teléfono, coordenadas en las que está situado el colegio y el código proporcionado por el colegio que certifica que es un colegio adherido al proyecto *Camino seguro al cole*. Una vez introducidos los datos correctamente pulsamos el botón *Aceptar* y el colegio quedará introducido en la base de datos.

### Insertar nuevo colegio



Nombre:  Ej: CEIP Rosalía de Castro  
Coordenadas:  Ej: 40.474375,-3.867187  
Dirección:  Ej: C/Rosalía de Castro 20  
Codigo:  Ej: 22222  
Telefono:  Ej: 123456789

Figura 28: Insertar nuevo colegio desde el panel de administración.

## Insertar calle en araña de movilidad

Desde este formulario podemos insertar una araña de movilidad para un colegio, para ello debemos introducirla calle a calle en la base de datos. Las calles se introducen con coordenadas de punto a punto, y se trazará una línea entre los dos puntos. Para introducir una calle en la araña de movilidad debemos conocer el par (latitud, longitud) de las coordenadas de origen y destino de los extremos de la calle. Para más información consultar el apartado *Obtención de la araña de movilidad de un colegio* a partir de las rutas subidas por los escolares de este mismo documento.

### Insertar calle en araña de movilidad



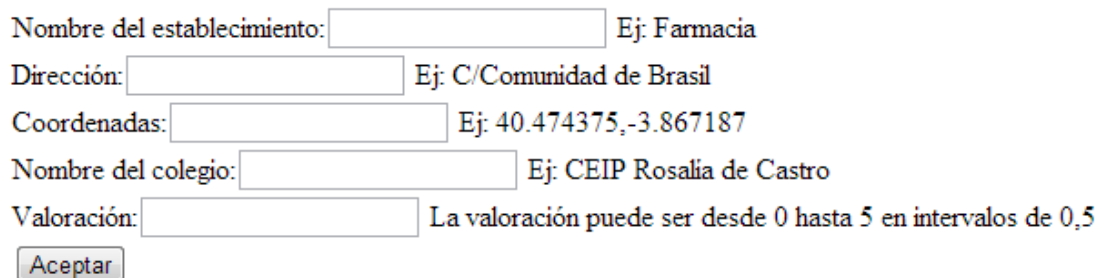
Nombre del colegio:  Ej: CEIP Rosalía de Castro  
Coordenada de origen:  Ej: 40.474375,-3.867187  
Coordenada destino:  Ej: 40.474375,-3.867187

Figura 29: Insertar calle en araña de movilidad desde el panel de administración.

## Insertar establecimiento

Podemos insertar un nuevo establecimiento amigo de los niños en la base de datos de la aplicación *Camino seguro al cole*. Para ello tendremos que rellenar los datos del formulario con el nombre del establecimiento, su dirección, las coordenadas en la que se encuentra y el nombre del colegio al que está asociado como parte del proyecto *Camino seguro al cole*. Opcionalmente se puede dar una valoración inicial del establecimiento rellenando el campo valoración. Una vez rellenados correctamente los campos y pulsando sobre el botón *Aceptar* se añade un nuevo establecimiento a la tabla establecimientos de la base de datos.

## Insertar establecimiento



Nombre del establecimiento:  Ej: Farmacia

Dirección:  Ej: C/Comunidad de Brasil

Coordenadas:  Ej: 40.474375,-3.867187

Nombre del colegio:  Ej: CEIP Rosalía de Castro

Valoración:  La valoración puede ser desde 0 hasta 5 en intervalos de 0,5

Figura 30: Insertar establecimiento desde el panel de administración.

## Obtener un registro con todas las rutas pertenecientes a un colegio

Desde esta sección se pueden recuperar las rutas creadas en la funcionalidad *Marcar mi camino al cole*, en la que cada niño sube a la base de datos el trayecto que sigue desde su casa hasta el colegio. Esta funcionalidad permite la recogida de los datos para la actualización de la araña de movilidad de una forma más cómoda que la actual (rellenando unos mapas en papel). Una vez obtenidas las rutas de todos los niños se procederá a la obtención de la araña de movilidad para cada colegio y a la posterior introducción de la araña de movilidad desde este mismo panel de administración.

Para obtener un registro con todas las rutas de un colegio sólo debemos introducir el nombre del colegio del cual queremos recopilar sus rutas y pulsar el botón *Aceptar*. Si el nombre introducido coincide con el de alguno de los colegios introducidos en la base de datos, se abrirá el gestor de descargas para guardar en nuestro disco un fichero Excel con las rutas punto a punto que cada usuario sigue desde su casa hasta el colegio.

## Obtener un registro con todas las rutas pertenecientes a un colegio.

Nombre del colegio:  Ej: CEIP Rosalía de Castro

*Figura 31: Obtención de un registro con todas las rutas para la actualización de la araña de movilidad de un colegio desde el panel de administración.*

## Obtener un registro con todos los colegios almacenados

Desde esta sección se puede realizar un volcado de la tabla colegios de la base de datos a un fichero Excel. Para ello sólo hay que pulsar sobre el botón *COLEGIOS* y el gestor de descargas se abrirá para seleccionar donde queremos guardar el archivo que contiene la lista de colegios adheridos al proyecto *Camino seguro al cole*.

## Obtener un registro con todos los colegios almacenados.

*Figura 32: Obtención de un registro con todos los colegios almacenados desde el panel de administración.*

## Obtener un registro con todos los establecimientos adheridos

Análogo a la obtención de un registro con todos los colegios almacenados, sólo hay que pulsar sobre el botón *ESTABLECIMIENTOS* y se abrirá el gestor de descargar para seleccionar donde queremos guardar el archivo Excel que contiene la lista de todos los establecimientos adheridos al proyecto *Camino seguro al cole*.

## Obtener un registro con todos los establecimientos adheridos.

*Figura 33: Obtención de un registro con todos los establecimientos amigos de los niños desde el panel de administración.*

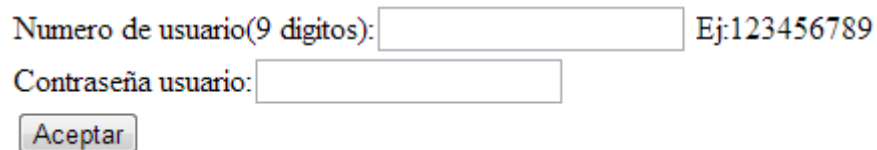
## Añadir nuevo administrador

Desde esta sección se puede añadir un nuevo administrador, proporcionándole acceso al panel de administración, desde el cual se puede gestionar la información contenida en la base de datos de la aplicación *Camino seguro al cole*.

Para añadir un nuevo administrador hay que rellenar los campos número de usuario y contraseña que habrá que proporcionar como identificación en la pantalla de inicio del panel de administración.

Una vez rellenado correctamente y pulsado el botón *Aceptar*, queda introducido en el sistema un nuevo administrador.

## Añadir nuevo administrador.



Formulario para añadir un nuevo administrador. Incluye dos campos de entrada: 'Numero de usuario(9 digitos):' y 'Contraseña usuario:'. El primer campo tiene un ejemplo 'Ej:123456789' a su derecha. Debajo de los campos hay un botón 'Aceptar'.

Numero de usuario(9 digitos):  Ej:123456789

Contraseña usuario:

*Figura 34: Añadir nuevo administrador desde el panel de administración.*



## Obtención de la araña de movilidad de un colegio a partir de las rutas subidas por los escolares.

1. Descargar e instalar la aplicación gratuita *Google Earth*.  
[http://www.google.es/intl/es\\_es/earth/download/ge/agree.html](http://www.google.es/intl/es_es/earth/download/ge/agree.html)

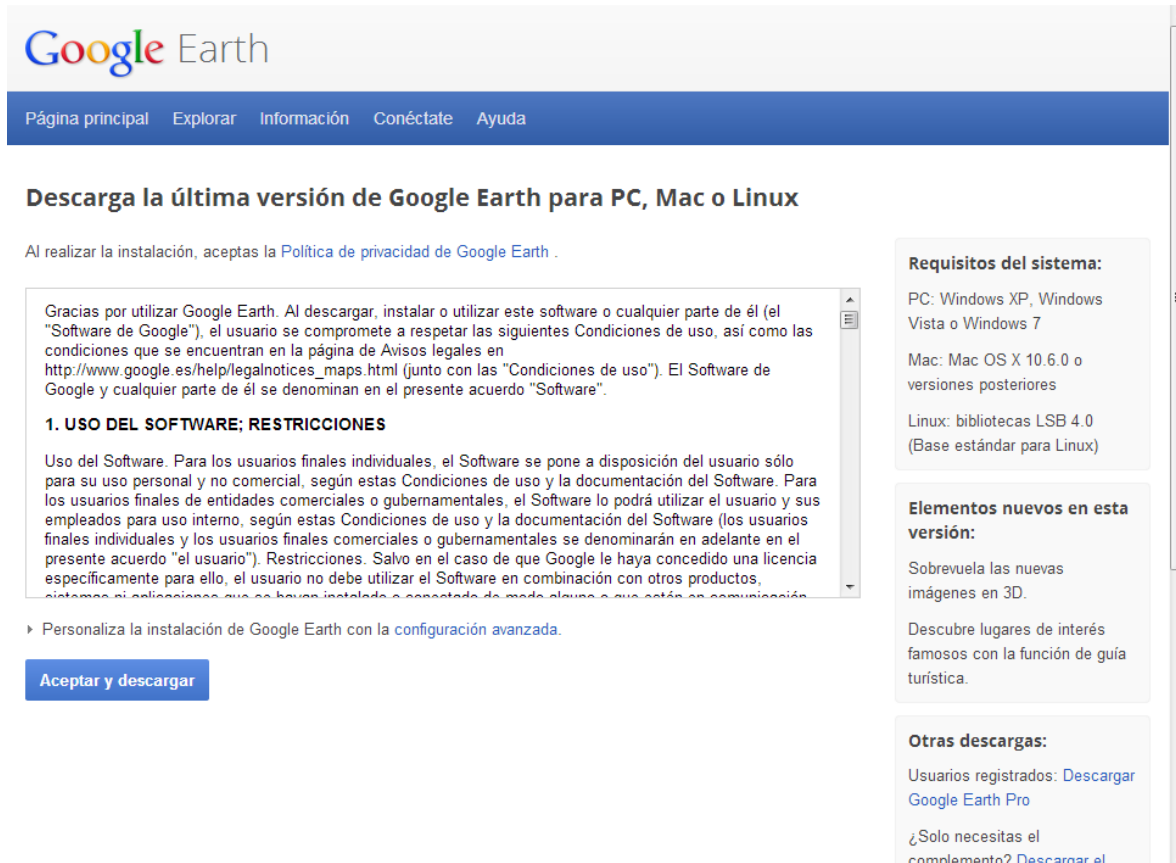


Figura 35: Descargar Google Earth para la obtención de la araña de movilidad.

2. Obtener todas las rutas de un colegio siguiendo los pasos de la sección *Obtener un registro con todas las rutas pertenecientes a un colegio* de este mismo manual.

- Una vez descargado el fichero abrir la carpeta contenedora y abrir el fichero. El fichero tiene el siguiente formato: El primer campo es el número de teléfono del alumno que subió la ruta.

El segundo campo es la lista de coordenadas en (latitud, longitud) que proporcionan la ruta seguida por el alumno punto a punto desde su casa hasta el colegio.

El tercer campo es el colegio al que pertenece dicha ruta.

	A	B	C
1	numUs	ruta	nombCol
2	685359552	lat/lng: (40.4902429,-3.8711507)lat/lng: (40.49164244513245,-3.868005573749542)lat/lng: (40.485948805967396,-3.8635078445076942)lat/lng: (40.48898399470453,-3.8580693304538727)lat/lng: (40.486558251668136,-3.850847482681274)lat/lng: (40.4836300567454,-3.8438298180699344)	CEIP Puerto Rico

Figura 36: Fichero con todas las rutas de un colegio para la obtención de la araña de movilidad de un colegio.

- Copiamos la ruta del primer alumno en un editor de texto para que sea más cómodo de leer y añadimos un salto de línea para cada par de coordenadas.

```

1 lat/lng: (40.4902429,-3.8711507)
2 lat/lng: (40.49164244513245,-3.868005573749542)
3 lat/lng: (40.485948805967396,-3.8635078445076942)
4 lat/lng: (40.48898399470453,-3.8580693304538727)
5 lat/lng: (40.486558251668136,-3.850847482681274)
6 lat/lng: (40.4836300567454,-3.8438298180699344)
7

```

Figura 37: Fichero con la ruta de un alumno para la obtención de la araña de movilidad de un colegio.

- Abrir la aplicación *Google Earth* desde el icono del escritorio.



Figura 38: Abrir la aplicación Google Earth.

6. Para tener mayor visibilidad deshabilitamos todas las capas que nos molesten en la parte inferior izquierda de la aplicación. Para volver a activarlas sólo hay que marcar la casilla otra vez. También podemos plegar la guía turística.

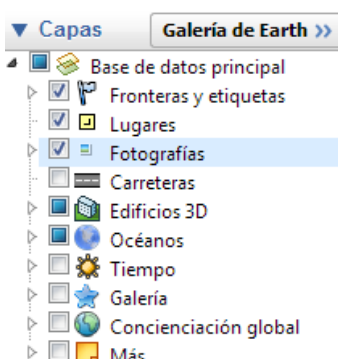


Figura 39: Deshabilitar capas en Google Earth.

7. Pulsamos sobre el botón *Añadir marcador* de la barra de herramientas de la parte superior de la aplicación.



Figura 40: Herramienta marcador de la aplicación Google Earth.

8. En el menú emergente, introducir la latitud y la longitud de la primera coordenada de la ruta subida por el primer alumno. La aplicación hace automáticamente la conversión de notación decimal a notación de grados, minutos y segundos. Como nombre podemos utilizar el número de teléfono desde el que se subió la ruta o cualquier otro nombre, por ejemplo alumno1.

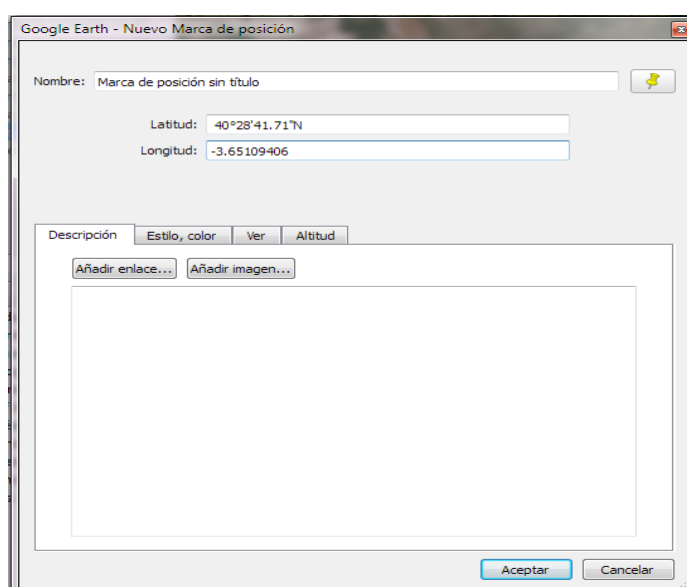


Figura 41: Añadir marcador.

9. Pulsar sobre el botón con la chincheta amarilla y cambiar el icono del marcador al icono con la letra A. De esta manera podemos diferenciar los puntos de la ruta de un alumno de los puntos de los demás alumnos. Seleccionaremos un marcador distinto para cada alumno.

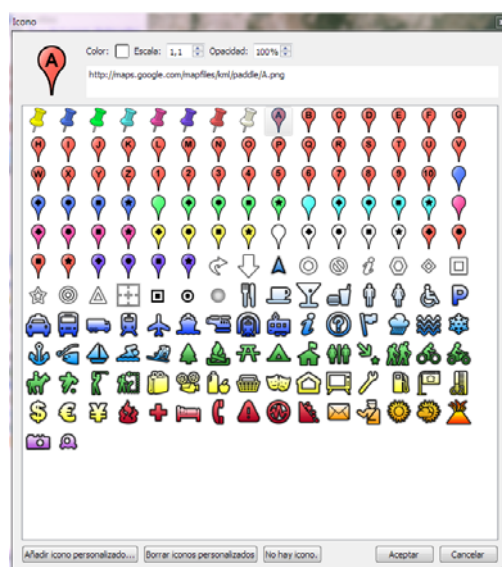


Figura 42: Seleccionar un marcador para los puntos de la ruta de un alumno.

10. Repetir desde el paso 7 para cada una de las coordenadas de la ruta del alumno. Una vez completado pasar al siguiente punto.
11. Buscamos el nombre del colegio en la barra de búsqueda de la aplicación e introducimos un marcador para la posición del colegio. Podemos seleccionar un icono especial para visualizarlo en el mapa

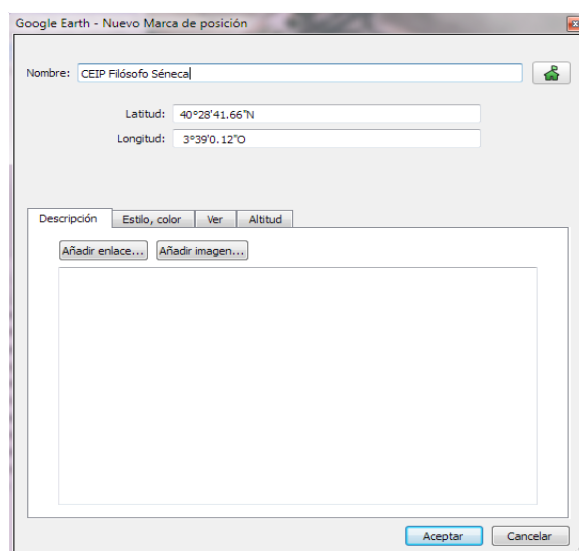
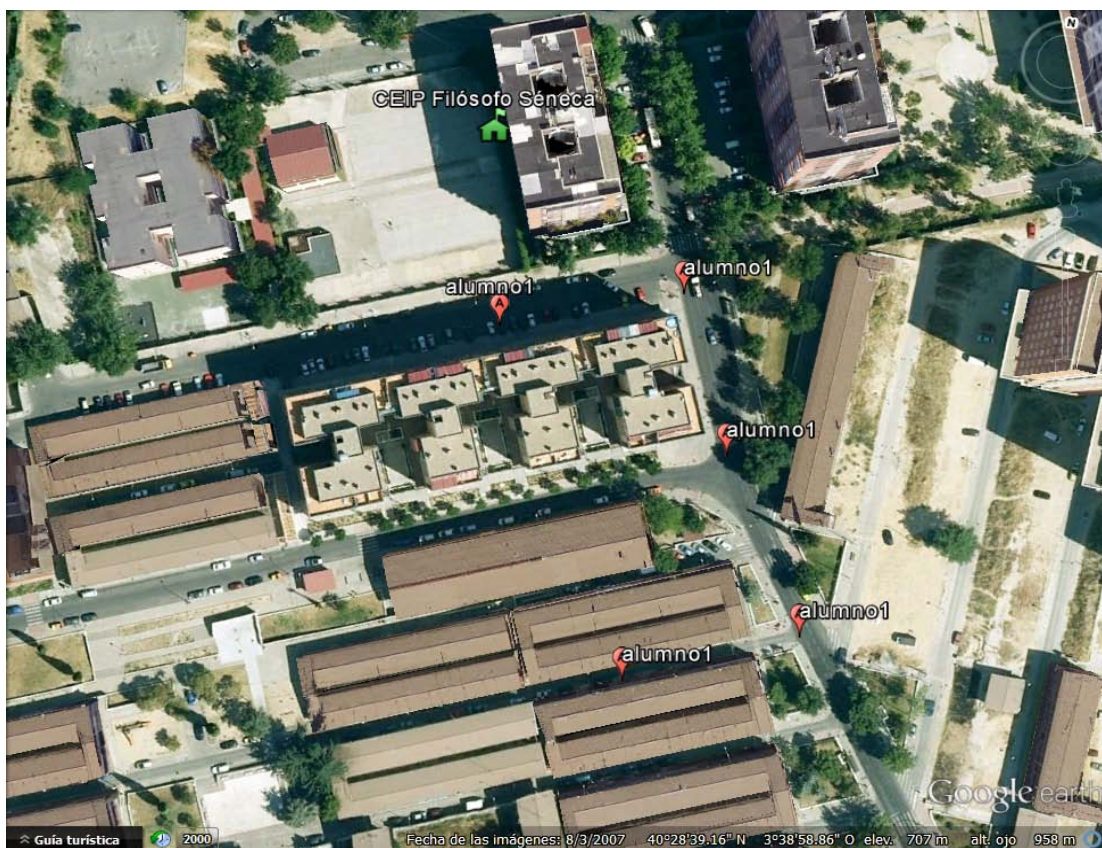


Figura 43: Añadir marcador para el colegio.

12. Una vez añadidos todos los marcadores al mapa fijamos el nivel de zoom apropiado para ver correctamente las calles y los marcadores.



*Figura 44: Marcadores de la ruta de un alumno.*

13. Seleccionamos la herramienta *Ruta* en la barra de herramientas de la parte superior de la aplicación.



*Figura 45: Herramienta ruta de la aplicación Google Earth.*



14. En el menú emergente ponemos como nombre el mismo nombre que hayamos usado para los marcadores y en la pestaña *Estilo, color* seleccionamos un color y fijamos el valor de la anchura a un valor que nos permita ver la línea cómodamente. Seleccionaremos un color diferente para la ruta de cada uno de los alumnos.

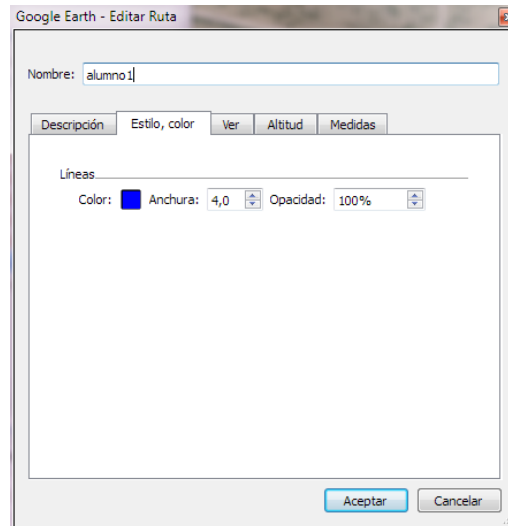


Figura 46: Selección de estilo y color para la ruta de un alumno

15. Ahora sólo tenemos que trazar la ruta sobre el mapa uniendo los puntos entre el colegio y los marcadores en el orden en que los hemos insertado. Si nos equivocamos podemos deshacer el último movimiento pulsando el botón derecho del ratón.

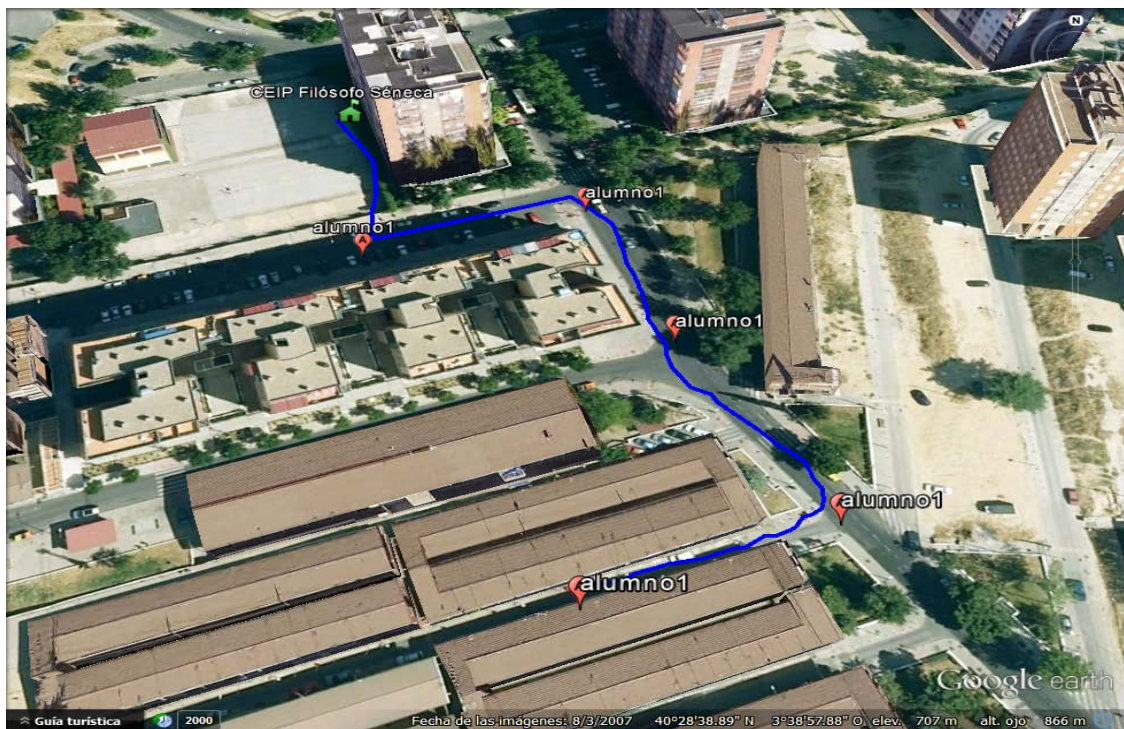


Figura 47: Creación de la ruta de un alumno.

16. Repetir desde el punto 7 para cada una de las rutas subidas por los alumnos. Ya podemos quitar los marcadores desmarcando las casillas de la pantalla *Lugares* en el panel izquierdo de la aplicación. Desde este panel podemos gestionar todas nuestras rutas y marcadores. Para eliminar un elemento sólo hay que seleccionarlo y pulsar la tecla suprimir.

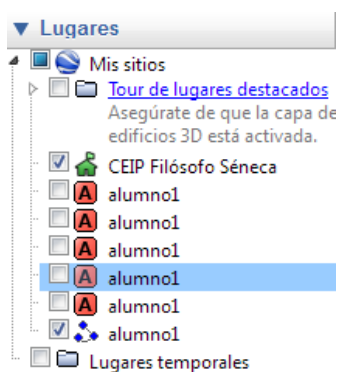


Figura 48: Deshabilitar marcadores.

17. Una vez recogidas todas las rutas se establecerá la araña movilidad para el colegio seleccionado, con el criterio que se considere (en este caso, la ruta por la que vayan el mayor número de niños).

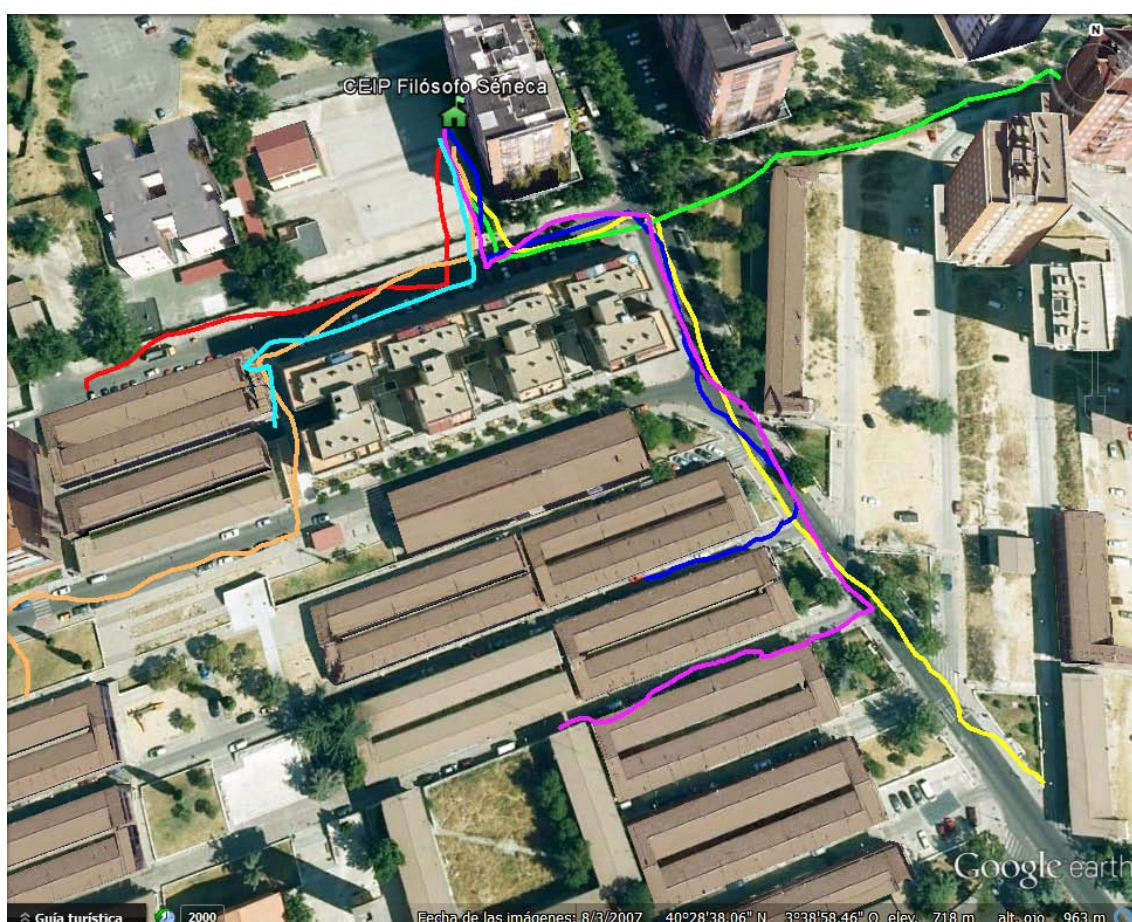


Figura 49: Conjunto de rutas de los alumnos de un colegio.



18. Fijar los puntos para la introducción de la araña de movilidad en la base de datos. Esto lo haremos añadiendo marcadores en los puntos de la araña de movilidad donde la línea cambie de dirección (sólo introducimos los puntos origen y destino de cada línea). Para ello pulsamos sobre el botón *Añadir marcador* de la barra de herramientas, arrastramos el marcador hasta el punto de cambio de dirección de la línea y le ponemos como nombre las coordenadas en latitud y longitud con un espacio de separación. Ya podemos deshabilitar las rutas de los alumnos desde el panel *Lugares*.

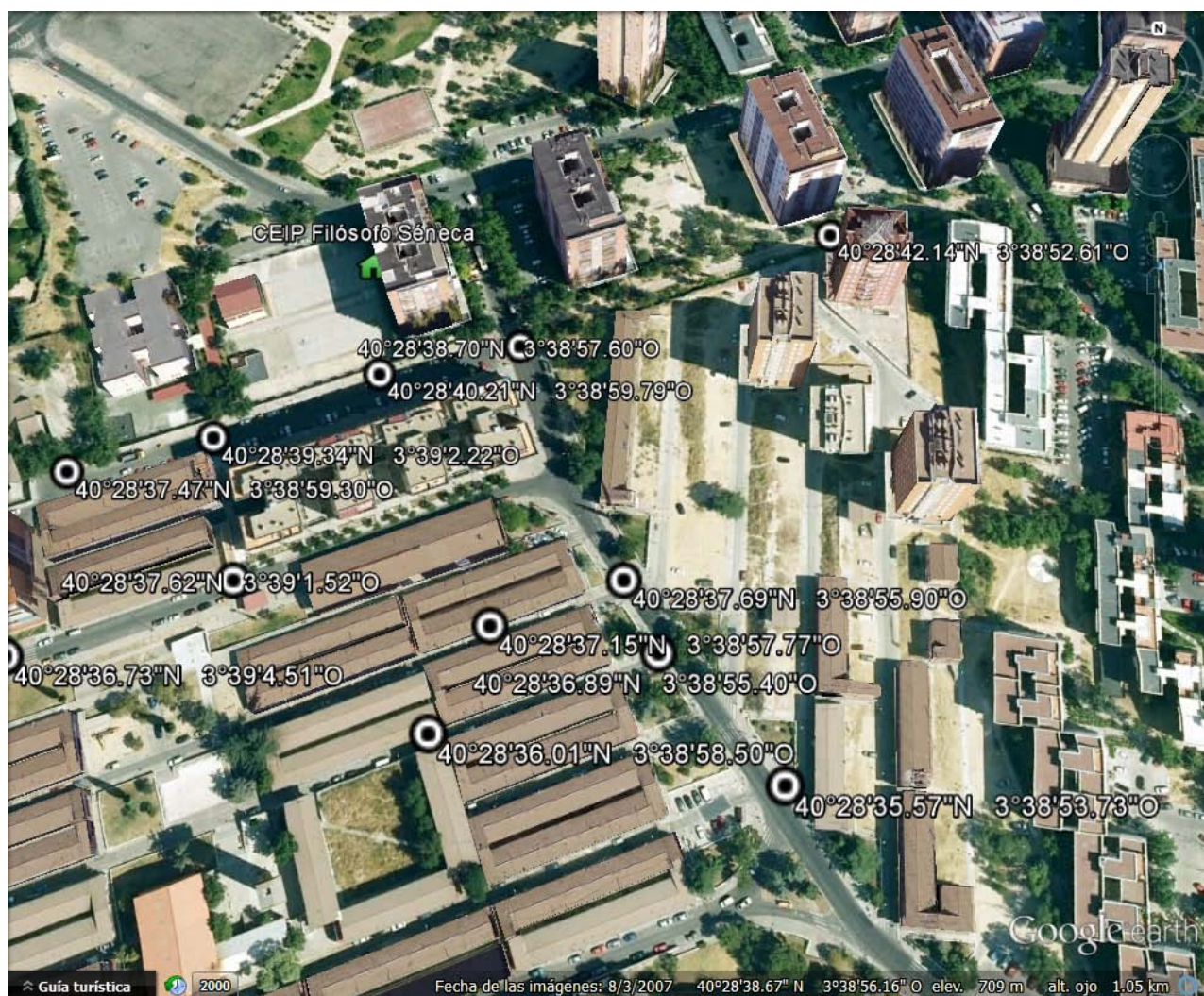


Figura 50: Representación punto a punto de la araña de movilidad de un colegio.



19. Convertir las coordenadas en notación de grados, minutos y segundos a notación decimal. Al hacer la conversión hay que tener en cuenta que si en la latitud aparece la letra 'N' el valor decimal será positivo, y si aparece la letra 'S' el valor decimal será negativo. En cuanto a la longitud, el valor decimal será positivo si aparece la letra 'E' y negativo si aparece la letra 'O'. Madrid está situada por encima del ecuador y a la izquierda del meridiano de Greenwich, luego la latitud siempre será positiva y la longitud siempre será negativa.

Para hacer la conversión se puede usar cualquier conversor o utilizar las funciones de Visual Basic<sup>[12]</sup> en Excel.

- Conversión de grados, minutos y segundos a grados decimales:

```
Function Convert_Decimal(Degree_Deg As String) As Double
    ' Declare the variables to be double precision floating-point.
    Dim degrees As Double
    Dim minutes As Double
    Dim seconds As Double
    ' Set degree to value before "" of Argument Passed.
    degrees = Val(Left(Degree_Deg, InStr(1, Degree_Deg, "") - 1))
    ' Set minutes to the value between the "" and the ""
    ' of the text string for the variable Degree_Deg divided by
    ' 60. The Val function converts the text string to a number.
    minutes = Val(Mid(Degree_Deg, InStr(1, Degree_Deg, "") + 2, _
        InStr(1, Degree_Deg, "") - InStr(1, Degree_Deg, _
            "") - 2)) / 60
    ' Set seconds to the number to the right of "" that is
    ' converted to a value and then divided by 3600.
    seconds = Val(Mid(Degree_Deg, InStr(1, Degree_Deg, "") + _
        2, Len(Degree_Deg) - InStr(1, Degree_Deg, "") - 2)) _
        / 3600
    Convert_Decimal = degrees + minutes + seconds
End Function
```

- Conversión de grados decimales a grados, minutos y segundos:

```
Function Convert_Degree(Decimal_Deg) As Variant
    With Application
        'Set degree to Integer of Argument Passed
        Degrees = Int(Decimal_Deg)
        'Set minutes to 60 times the number to the right
        'of the decimal for the variable Decimal_Deg
        Minutes = (Decimal_Deg - Degrees) * 60
        'Set seconds to 60 times the number to the right of the
        'decimal for the variable Minute
        Seconds = Format(((Minutes - Int(Minutes)) * 60), "0")
        'Returns the Result of degree conversion
        '(for example, 10.46 = 10~ 27 ' 36")
        Convert_Degree = " " & Degrees & "° " & Int(Minutes) & "'
    " -
        & Seconds + Chr(34)
    End With
End Function
```

20. Introducir cada pareja de puntos origen-destino desde la *sección Insertar calle en araña de movilidad* del panel de administración. De esta forma queda introducida en la base de datos la araña de movilidad para el colegio seleccionado, y podrá consultarse en el botón *Mi ruta segura* de la aplicación *Camino seguro al cole*.

## Anexo 6. Registro de reuniones de especificación

**Fecha:** miércoles 19 de septiembre.

**Hora:** 11:00.

**Lugar:** Aula 13 de la facultad de informática.

**Asistentes:** Victoria López, Guadalupe Miñana, Inma Pardines e integrantes de los grupos *Hábitat Madrid*, *Itinerarios del Retiro*, *Mapa de recursos*, *Recycla.me*, *Recycla.te* y *Camino seguro al cole*.

**Temas tratados:** reunión general para ver las normativas y el método de trabajo para la asignatura sistemas informáticos. Convocatoria de reuniones específicas para cada grupo.

**Fecha:** jueves 11 de octubre.

**Hora:** 09:20.

**Lugar:** calle Montalbán, 1 – 6ª planta. Área de gobierno de medioambiente, seguridad y movilidad. Ayuntamiento de Madrid.

**Asistentes:** Victoria López, Guadalupe Miñana, Marisol Mena, Nines Toribio e integrantes de los grupos *Hábitat Madrid*, *Itinerarios del Retiro*, *Mapa de recursos*, *Recycla.me*, *Recycla.te* y *Camino seguro al cole*.

**Temas tratados:** reunión general para fijar las bases de los proyectos. Primera especificación de la herramienta.

**Fecha:** viernes 26 de octubre.

**Hora:** 11:30.

**Lugar:** sala de reuniones de la facultad de informática.

**Asistentes:** Victoria López, Álvaro Bustillo, Arturo Callejo, Héctor Martos.

**Temas tratados:** especificación de la interfaz y diagrama de flujo de la aplicación.

**Fecha:** miércoles 31 de octubre.

**Hora:** 09:00.

**Lugar:** facultad de matemáticas.

**Asistentes:** Victoria López, Inés Gómez, Rocío Colina, Beatriz Municio, Álvaro Bustillo, Héctor Martos.

**Temas tratados:** presentación proyectos matemáticos de optimización de rutas y comparación de rutas con transportes alternativos en el *Camino seguro al cole*.

**Fecha:** miércoles 31 de octubre.

**Hora:** 15:00.

**Lugar:** aula 13 de la facultad de informática.

**Asistentes:** Victoria López, Guadalupe Miñana, Inma Pardines e integrantes de los grupos *Hábitat Madrid*, *Itinerarios del Retiro*, *Mapa de recursos*, *Recycla.me*, *Recycla.te* y *Camino seguro al cole*.

**Temas tratados:** estructura para la memoria del proyecto de fin de carrera, problemas y dudas con las especificaciones y análisis de requisitos.

**Fecha:** viernes 16 de noviembre.

**Hora:** 10:30.

**Lugar:** calle Montalbán, 1 – 6ª planta. Área de gobierno de medioambiente, seguridad y movilidad. Ayuntamiento de Madrid.

**Asistentes:** Victoria López, Marisol Mena, Nines Toribio, Jesús Escribano, Rocío Colina, Beatriz Municio, Álvaro Bustillo, Héctor Martos.

**Temas tratados:** problemas con la primera especificación de la herramientas. Cambios en la funcionalidad de la aplicación. Segunda especificación de la herramienta.

**Fecha:** miércoles 5 de diciembre.

**Hora:** 15:00.

**Lugar:** aula 13 de la facultad de informática.

**Asistentes:** Victoria López, Guadalupe Miñana, Inma Pardines e integrantes de los grupos *Hábitat Madrid*, *Itinerarios del Retiro*, *Mapa de recursos*, *Recycla.me*, *Recycla.te* y *Camino seguro al cole*.

**Temas tratados:** reunión general para la preparación de presentaciones sobre las aplicaciones, explicación del capítulo de la memoria estado del arte, organización de la carpeta en Dropbox.

**Fecha:** lunes 14 de enero.

**Hora:** 15:30.

**Lugar:** despacho de Victoria López

**Asistentes:** Victoria López, Álvaro Bustillo, Arturo Callejo, Héctor Martos.

**Temas tratados:** revisión del estado general del proyecto (aplicación y memoria).

**Fecha:** miércoles 6 de febrero.

**Hora:** 15:30.

**Lugar:** aula 13 de la facultad de informática.

**Asistentes:** Victoria López, Guadalupe Miñana, Inma Pardines e integrantes de los grupos *Hábitat Madrid*, *Itinerarios del Retiro*, *Mapa de recursos*, *Recycla.me*, *Recycla.te* y *Camino seguro al cole*.

**Temas tratados:** reunión general para ver el estado de los proyectos.

**Fecha:** viernes 15 de febrero.

**Hora:** 15:30.

**Lugar:** aula 12 de la facultad de informática.

**Asistentes:** Victoria López, Guadalupe Miñana, Inma Pardines e integrantes de los grupos *Hábitat Madrid*, *Itinerarios del Retiro*, *Mapa de recursos*, *Recycla.me*, *Recycla.te* y *Camino seguro al cole*.

**Temas tratados:** comentarios sobre la guía de estándares del IAM.

**Fecha:** viernes 22 de febrero.

**Hora:** 11:15.

**Lugar:** sala de reuniones de la facultad de informática.

**Asistentes:** Victoria López, Jesús Escribano, Álvaro Bustillo, Arturo Callejo, Héctor Martos.

**Temas tratados:** listado de documentación para solicitar al Ayuntamiento (arañas de movilidad y comercios amigos de los niños), comentarios del código y solicitud de reunión con el Ayuntamiento.

**Fecha:** lunes 4 de marzo.

**Hora:** 10:00.

**Lugar:** calle Montalbán, 1 – 6ª planta. Área de gobierno de medioambiente, seguridad y movilidad. Ayuntamiento de Madrid.

**Asistentes:** Victoria López, Marisol Mena, Nines Toribio, Rocío Colina, Beatriz Municio, Álvaro Bustillo, Arturo Callejo, Héctor Martos.

**Temas tratados:** presentación del primer prototipo de la aplicación *Camino seguro al cole* y de los proyectos matemáticos de optimización de rutas y comparación de rutas con transportes alternativos en el *Camino seguro al cole*.

**Fecha:** viernes 15 de marzo.

**Hora:** 09:00.

**Lugar:** colegio de enseñanza infantil y primaria Filósofo Séneca.

**Asistentes:** Marisol Mena, Nines Toribio, directora del colegio, padres en representación del AMPA, Héctor Martos.

**Temas tratados:** presentación del prototipo de la aplicación *Camino seguro al cole*.

**Fecha:** viernes 12 de abril.

**Hora:** 16:00.

**Lugar:** sala de reuniones de la facultad de informática.

**Asistentes:** Victoria López, Jesús Escribano, Álvaro Bustillo, Arturo Callejo, Héctor Martos.

**Temas tratados:** comentarios sobre la última versión de la aplicación.

**Fecha:** lunes 22 de abril.

**Hora:** 15:30.

**Lugar:** aula 7 de la facultad de informática.

**Asistentes:** Victoria López, Guadalupe Miñana, Inma Pardines e integrantes de los grupos *Hábitat Madrid*, *Itinerarios del Retiro*, *Mapa de recursos*, *Recycla.me*, *Recycla.te* y *Camino seguro al cole*.

**Temas tratados:** primer ensayo de la presentación de las aplicaciones en la Catedral Innova. Recomendaciones generales y específicas.

**Fecha:** viernes 26 de abril.

**Hora:** 09:30.

**Lugar:** calle Montalbán, 1 – 6ª planta. Área de gobierno de medioambiente, seguridad y movilidad. Ayuntamiento de Madrid.

**Asistentes:** Victoria López, Jesús Escribano, Marisol Mena, Nines Toribio, Álvaro Bustillo, Arturo Callejo, Héctor Martos.

**Temas tratados:** presentación de la última versión de la aplicación. Cambios en los colores de la interfaz según la guía de diseño de Android. Refinamiento de la especificación.

**Fecha:** martes 7 de mayo.

**Hora:** 15:00.

**Lugar:** aula 7 de la facultad de informática.

**Asistentes:** Victoria López, Guadalupe Miñana, Inma Pardines e integrantes de los grupos *Hábitat Madrid*, *Itinerarios del Retiro*, *Mapa de recursos*, *Recycla.me*, *Recycla.te* y *Camino seguro al cole*.

**Temas tratados:** segundo ensayo de la presentación de las aplicaciones en la Catedral Innova. Últimas recomendaciones.

**Fecha:** viernes 17 de mayo.

**Hora:** 10:30.

**Lugar:** calle Monforte de Lemos, 58. Centro de innovación Vaguada.

**Asistentes:** Victoria López, Guadalupe Miñana, Inma Pardines e integrantes de los grupos *Hábitat Madrid*, *Itinerarios del Retiro*, *Mapa de recursos*, *Recycla.me*, *Recycla.te* y *Camino seguro al cole*.

**Temas tratados:** presentación de las aplicaciones para medio ambiente y movilidad en el centro de innovación Vaguada.

**Fecha:** miércoles 23 de mayo.

**Hora:** 12:00.

**Lugar:** plaza de Cibeles, 1. Edificio Centro-Centro Ayuntamiento de Madrid.

**Asistentes:** Victoria López, Guadalupe Miñana, Inma Pardines, Marisol Mena, Nines Toribio, Isabel Pachón, José Luis, María Fúster e integrantes de los grupos *Hábitat Madrid*, *Itinerarios del Retiro*, *Mapa de recursos*, *Recycla.me*, *Recycla.te* y *Camino seguro al cole*.

**Temas tratados:** presentación de las aplicaciones para medio ambiente y movilidad en la sala “Sigfrido Martín Begue”.